电脑

3
1987

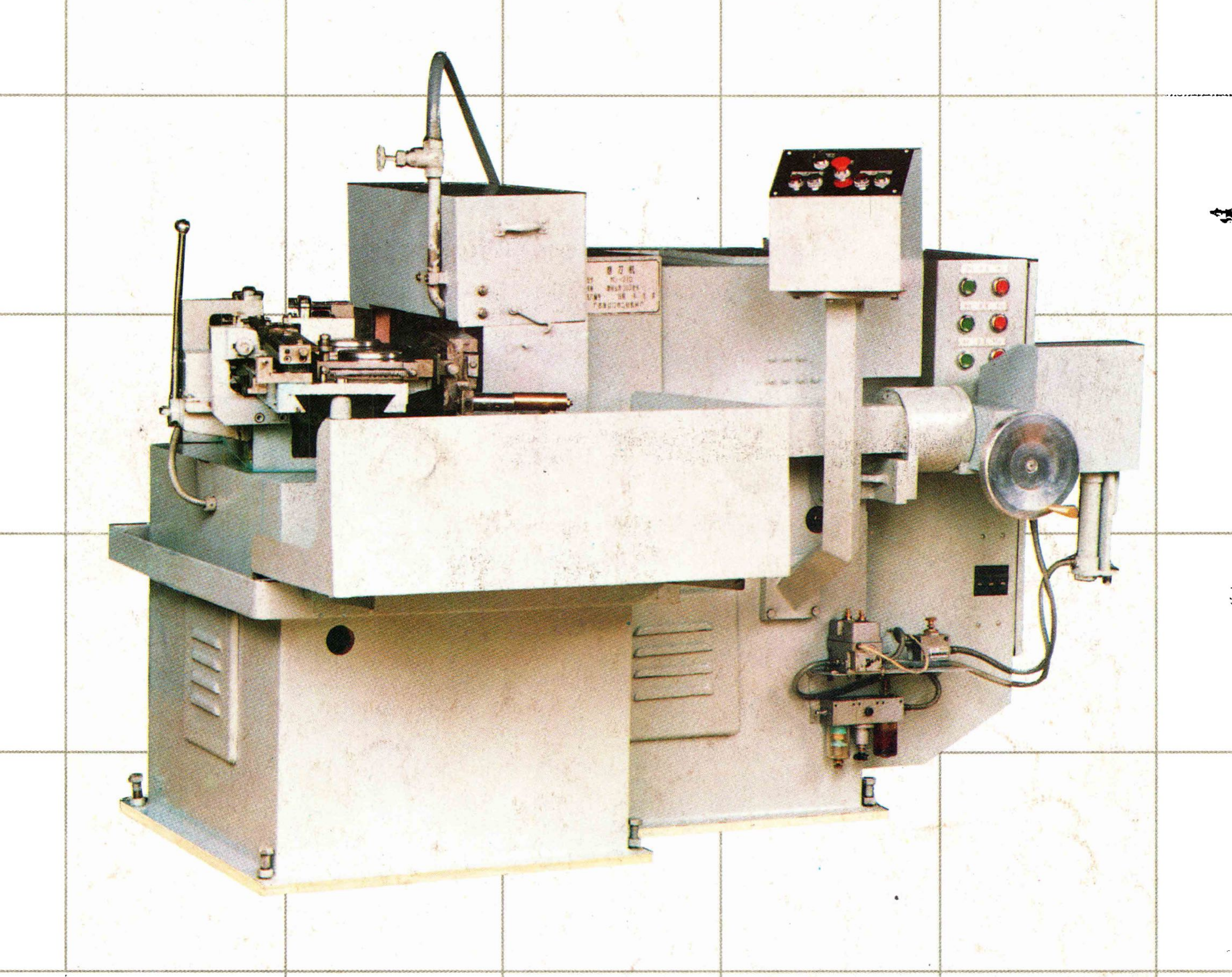
中国软件行业协会会刊

MD－310磨刀机

（该机分左右向）

MD–310 Cutlery–edge Crinder

(option of right–side or left–side grinding)



江门市二轻机械厂

厂址：广东省江门市胜利路134 号

电话：33850　　　　电挂：9222

Jiangmen Machinery Works of The Second Light Industry

Telephone : 33850

Cable :　　9222

Address :　134 Shengli Road Jiangmen Guangdong People's Republic of China

京粤汉字电脑中心生产基地



基地地址：广州市三元里棠溪南路八号　电话：627127　627141

京粤汉字电脑技术研究开发中心（BGC）是由中国科学院计算技术研究所、财政部计算中心、广东省计算技术应用研究所合作组成的一个联合实体。

“京粤中心”将电脑的研究、设计、开发、生产、应用、维修服务有机地结合在一起，充分发挥各联合单位的优势，为中国计算机的国产化、汉字化做出了贡献。

“京粤中心”采用现代化的管理方法和技术，具有较高的工作效率和经济效益。

“京粤中心”有一支强大的科研队伍，承担着研究、设计和开发的任务，中、高级科研人员近100人，在系统设计、系统实现和应用开发等方面都有一批具有实际经验的研究人员和工程技术人员，并聘请了近二十名计算机界知名的专家、学者任顾问。

“京粤中心”有自己的生产基地、电装、机装、测试、调试、考机等各个工艺环节已初具规模，能独立完成国内外委托的各种计算机系统分析，设计和实现，以及微型机的开发应用，人才培训，技术咨询和技术服务等各项任务。

“京粤中心”设计、研究、生产的汉字微机系统已形成京粤GF 系列化产品，已推出的GF 20　11A Ⅰ型、Ⅱ型、Ⅲ型已经广泛使用，刚推出的京粤GF 0520C－H、京粤GF 0520汉字通用终端、京粤GF 0520通用打印机汉字智能控制器和京粤GF 0530分布式汉字微机系统、京粤GF 0530网络，深受欢迎。

京粤GF 0520微机曾四次获得省、部级及国家级奖励、曾六度被省及国家选送出国赴港、日、美、加、法等地展览。

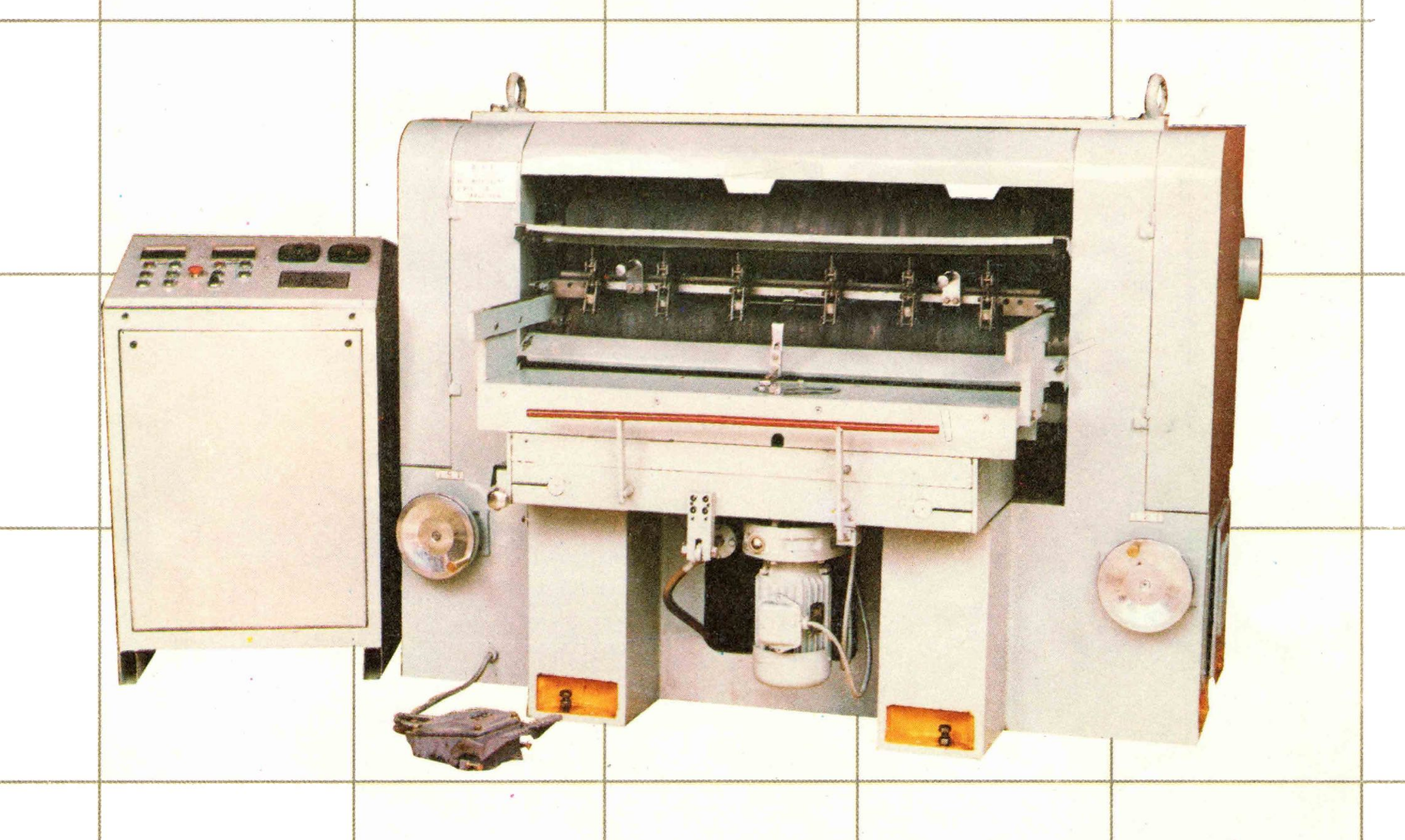
目前，京粤GF 0520产品已在全国财税系统、石油、银行、机关和科研单位等领域推广应用，并将进一步扩大应用到邮电、公安、教育、组织、人事、商业、体育、旅游部门。

总部地址：广州市连新路科学馆大院　电话：627127　341154　电挂：5374　电传：44676 GSTSCCN

分部地址：北京市中关村中国科学院计算所内　电话：283131—587　电挂：4615

P—1062 抛光机

P—1062 Polishing Machine



江门市二轻机械厂

广址：广东省江门市胜利路134 号

电话：33850　　电挂：9222

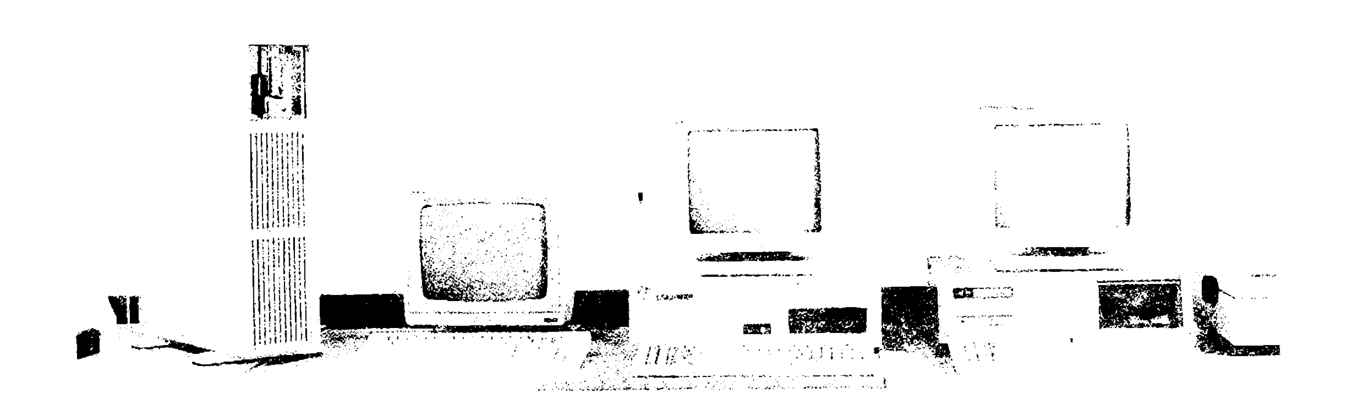
Jiangmen Machinery Works of The Second Light Industry

Telephone: 33850

Cable: 9222

Address: 134 Shengli Road Jiangmen Guangdong People's Republic of China

京粤GF0500 系列微机产品简介

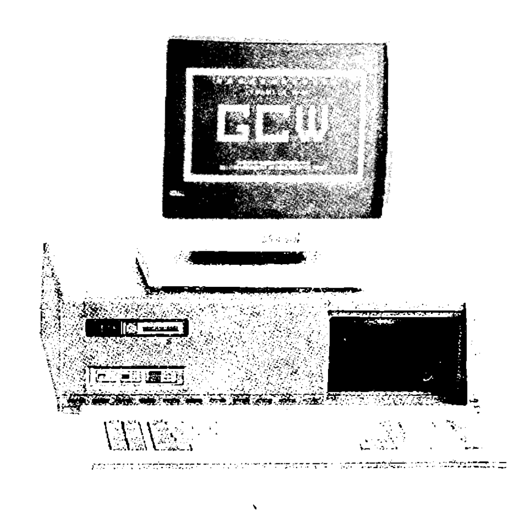


▲ 京粤GF 0530〔GF 20　11 B〕分布式微机系统

双CPU：Intel 80186、Z 80A　　磁　带：60 MB

内　存：512 KB～1 MB　　软　盘：640 KB

硬　盘：50～140 MB　　可连16个用户机，运行多用户数据库等软件



◀ 京粤GF 0520 C—H〔GF 20　11 C〕

CPU：8088且或80286

内　存：640 KB

硬　盘：33 MB

软　盘：360 KB

主　频：8 MHz

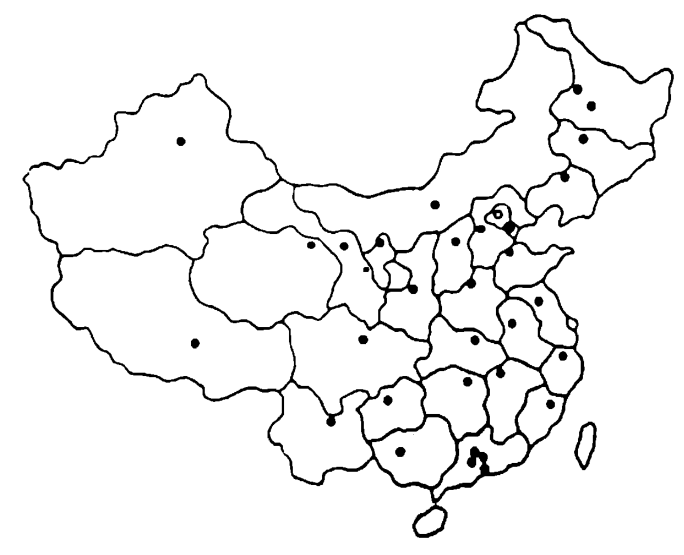
可打印出仿宋、楷体、黑体等39种字号的汉字字体

可作为京粤GF 0530的用户机

※京粤还为您提供如下软件：

- ●实用文书处理系统（PDS）
- ●文件报告管理系统（WJGL）
- ●汉字屏幕编辑软件（HZBJ）
- ●数字化仪绘图软件（HTRJ）
- ●中医针灸医疗系统
- ●中医保健系统
- ●电信业务信息管理系统（TIMS）
- ●字元词码
- ●五笔型
- ●COBOL 1.0
- ●C 语言1.04
- ●通用税票处理系统（GTPS）
- ●财税资料查询系统（GTRS）
- ●机关通用财务管理软件（GCW）
- ●高级制表软件（HRPG）
- ●通用汉字报表系统（GCRS）
- ●中国食谱精选
- ●用户建词、建编码程序
- ●用户词组
- ●FORTRAN 1.01
- ●PASCAL 1.31
- ●dBASE Ⅱ

京粤汉字电脑　产品遍布全国
四次国内获奖　六度出国展出





京粤电脑在香港

◀ 荣获诺贝尔奖的著名物理学家杨振宁博士观看了京粤电脑表演后，连声称赞："很好！很好！"。



▲京粤电脑曾多次获奖

京粤电脑在美国、加拿大

▶ 在美国国际电脑展览会上，引起了巨大反响。



京粤电脑在日本

◀ 在日本的筑波国际科技博览会上，受到广泛的瞩目。



电脑

（月　刊）

1987 年

第三期（总第15期）

厂家与产品

责任编辑：曾令宝 吴 军
美术编辑：丘文东

主 办 电子工业部计算机工业管理局
编 辑 《电脑》编辑部
（地址：广州市石牌华南师范大学微电子学研究所）
出 版 华南师范大学微电子学研究所
排 版 电脑杂志社电脑激光排版中心
印 刷 佛山市禅城印刷厂
总发行处 韶关市邮电局
订 阅 处 全国各地邮电局、所
刊 号 46－115
定 价 0.50元
出版日期 87年3月1日
广东省期刊登记证第118号
广告经营许可证穗工商广字004号

C－dBASEⅢ使用中的一种小技巧

怀化市国营4507厂 孙 侠

使用 C－DBASEⅢ 编制应用程序时，发现在使用删除命令时，有点麻烦，即：

delete recno() 加删除标记
pack 物理删除

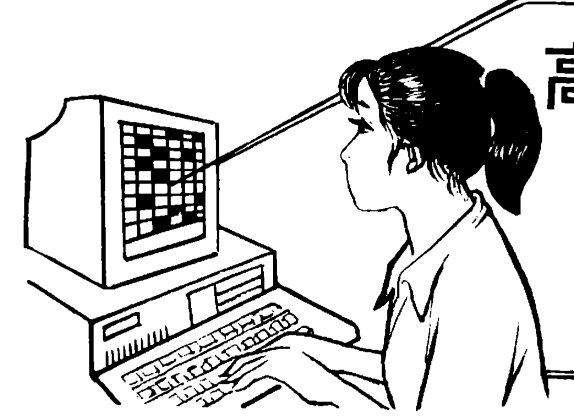
但执行删除命令后，花费的时间比较长，如一个数据库有2000条记录，要删除一个记录，将要5分钟。如果经常执行删除命令，那么花费的时间就比较惊人了。能否不用删除命令而达到要求呢？笔者在我厂的微机仓库管理中，试用了一种方法，效果较好。

我们的数据库一般都建立有索引文件，如果某记录的索引字段为空，那么这条记录就会排在索引文件的首位。基于这样一个道理，我们要删除一个记录，就不必用删除命令了，而只需将索引字段置空字符串，将数字字段置零就行了。然后每输入一次记录，先判断一下首记录的索引字段是否为空符，如是就用新记录将其覆盖；如不是新记录追加写入数据库尾。

程序是设计成这样的：

```
1．删除记录
   Use 文件名 Index 索引文件名
    [找到要删除的记录]
     Replace 索引字段名 With 索引字段名等长的空符，;
             数字字段   With  0.00
  Use
2．输入记录
   Use 文件名 Index 索引文件名
    [输入变量]
    If len(trim(索引字段名))=0
     Replace 字段名 with 变量，()with()…
    Else
     Append blank
      replace 字段名 With 变量，()With()…
    Endif
   Use
```

这样，删除记录比使用删除命令所需的时间就少多了，从而达到了节约主机运行时间的目的。



高解析度图形剪辑程序

陈志军

目前APPLE Ⅱ的游戏软件成千上万，这些软件中有着十分丰富多采的高解析度图形资源，把它们收集起来，按内容可以编成一个图形库。本程序的用途是利用图形库资料，进行剪贴、编辑，加工成为我所用的、面目一新的高解析度图形。

㈠程序的使用

这个名为APPLE MAKEUP的程序，首先给你提供三种功能选择：

1·是否要从源图形中剪裁出一部份图形，贴到（Y）新图上面；或将全图进行编辑（N）。

2·是否将图形上下颠倒过来。

3·是否贴成左右对称的镜象图形。

这三种功能可组合成八种不同的剪辑形式。

接着要求你输入源图形的文件名，输入被编辑或被粘贴的目标图名，若用空白的版面可按<RETURN>键而不输入图形名字。这时程序将存放于磁盘的源图形读入第一页高解析度图区并显示出来。为便于剪辑，图的左边缘按每5行画上一条刻度线，又在图下方标上0至40的列标号码。

接下来提供输入有关位置的数据：

1·按水平方向剪取从第N1列至第N2列范围内的图形，以及按垂直方向剪取第M1行至M2行范围内的图形。

2·把剪取的图形贴到第HTAB列和第VTAB行为起始位置的版面上。

上面每项输入均提示选值的范围供参考。全图形编辑无须输入这些数据，会跳过这部分内容。

位置数据决定后，程序读人目标图形到第二页图区并显示出来，紧跟着在这图形上（或空白图版上）绘图，绘图完毕程序转入文字幕，只要你给新编辑的图形命名，便存放入磁盘供你使用或再剪辑

㈡程序的说明

本程序第115句是给第一页图象的左边刻度。为此牺牲了第一列图形，如果这列位置的图形正合用，可以取消这一句，并将第140句中N1的初值从1改为0。

第170句是提示HTAB的值（H）取在该范围内，图形会迭加起来产生奇妙的效果。

鉴于高解析度图形与内存地址在垂直方向是不连续的，第240句至第440句进行的是图形——地址——图形的换算。设某点坐标为X，Y，该点内存地址为A，图址关系式是：

$$A=8192+40\times P+128\times Q+1024\times M+N$$

各数据关系是：N=INT（X/7），V（行）=INT（Y/8），P=INT（V/8），Q=V−8×P，M=Y−V×8，图形划分为三段（P），每段包括0至7个区（Q），每区又由0至7个单元组成（M），每行有40单元，每列有24单元。这部分程序是本程序的核心部分。

其中第270句将图上某点的地址求出来，若是暗点则跳过循环内容，提高绘图速度。第280句至300句是解除跨段（P）时在本段内卷绕的判断语句。第330句与340句使图象绘至右边、

下边时卷绕回左边、上边的图幅去。第３５０、３６０、４１０句是求出三种选择时新地址的值。第３８０句至第４１０句是将每行中每个字元的十进制值化为二进制值，并求余后再化为十进制数，产生镜象图形来。

㈢程序的改进

本程序用输入数据定位法进行剪贴比较麻烦，若改用操纵光标在图中移动，进行读取定位值的方法，读入剪裁范围的初值与终值，指示粘贴位置的始值，使用上会更方便。这样做要解决光标定位和彩色图象上颜色的还原问题，程序将较长。

另外这里用ＢＡＳＩＣ语言编程、又用了大量判断语句，绘图速度相当慢。可用ＭＩＣＲＯＳＯＦＴ——ＴＡＳＣ软件把它改编成机器语言的程序，或将绘图部分改编为机器语言子程序，用ＢＡＳＩＣ语言的主程序调用，速度将增加数倍。

有兴趣的朋友不妨动手试一试！

```
10   TEXT : HOME :D$ =  CHR$ (4): ONERR
       GOTO 10
20   HTAB 6: PRINT "******** APPLE
       MAKEUP ********": PRINT
30   PRINT "PICTURE......"
40   PRINT  TAB( 10);"CLIP AND PAS
      TE?(Y/N)    ";: GET X$: PRINT
      X$
50   PRINT  TAB( 10)"TURN UPSIDE D
      OWN?(Y/N) ";: GET Y$: PRINT
      Y$
60   PRINT  TAB( 10)"LOOKING-GLASS
      ?(Y/N)     ";: GET Z$: PRINT
      Z$
70   INPUT "ENTER NAME(SOURCE): ";
      A$
80   HTAB 7: INPUT "NAME(OBJECT) O
      R <RETURN>:";B$
90   HGR : VTAB 21: FOR I = 1 TO 4
      : FOR J = 0 TO 9: PRINT J;: NEXT
      J,I: PRINT
100  POKE 34,21: VTAB 22: PRINT "
      --------  TABLE  --------"
110  PRINT D$"BLOAD "A$",A$2000"
115  POKE 8192,127: POKE 8320,127
      : POKE 8960,127: POKE 8616,1
      27: POKE 8272,127: POKE 8912
      ,127
120  IF X$ = "N" THEN N1 = 0:N2 =
      39:M1 = 0:M2 = 23:H = 0:V =
      0: IF Z$ = "Y" THEN N2 = 19
130  IF X$ = "N" THEN 190
140  INPUT "START(N1),END(N2) [1-
      39]:";N1,N2: IF N1 < 1 OR N2
       > 39 OR N2 < N1 + 1 THEN 14
      0
150  INPUT "START(M1),END(M2) [0-
      23]:";M1,M2: IF M1 < 0 OR M2
       > 23 OR M2 < M1 + 1 THEN 15
      0
160  PRINT "HTAB(0-";39 - N2 + N1
      ;"):";: INPUT " ";H: IF H <
      0 OR H + N2 - N1 > 39 THEN 1
      60
170  IF Z$ = "Y" THEN  PRINT "***
      * IF "19 - N2 + N1"<H<"19 +
      N2 - N1" THEN PYRAMID! ****"
180  PRINT "VTAB(0-";23 - M2 + M1
      ;"):";: INPUT " ";V: IF V <
      0 OR V + M2 - M1 > 23 THEN 1
      80
190  IF Y$ = "Y" THEN V = 23 - V -
      M2 + M1
200 V = V - M1: IF V < 0 THEN V =
      24 + V
210 SP =  INT (M1 / 8):SQ = M1 -
      8 * SP:EP =  INT (M2 / 8):EQ
       = M2 - 8 * EP
220  HGR2 : IF B$ <  > "" THEN  PRINT
      D$"BLOAD "B$",A$4000"
230  POKE  - 16304,0: POKE  - 162
      97,0: POKE  - 16299,0
240  FOR P = SP TO 2: IF P <  > S
      P THEN SQ = 0
250  FOR Q = SQ TO 7: FOR M = 0 TO
      7: FOR N = N1 TO N2
260  IF P = EP AND Q = EQ THEN 45
      0
270 A0 = 8192 + 40 * P + 128 * Q +
      1024 * M + N: IF  PEEK (A0) =
      0 THEN 440
280  IF Q + V > 23 THEN P1 = P +
      3:Q1 = Q + V - 24: GOTO 320
290  IF Q + V > 15 THEN P1 = P +
      2:Q1 = Q + V - 16: GOTO 320
300  IF Q + V > 7 THEN P1 = P + 1
      :Q1 = Q + V - 8: GOTO 320
310 Q1 = Q + V:P1 = P
320 NV = N - N1 + H
330  IF NV > 39 THEN NV = NV - 40
340  IF P1 > 2 THEN P1 = P1 - 3
350  IF Y$ <  > "Y" THEN A1 = 819
      2 + A0 + 40 * (P1 - P) + 128
       * (Q1 - Q) + NV - N
360  IF Y$ = "Y" THEN A1 = 16384 +
      40 * (2 - P1) + 128 * (7 - Q
      1) + 1024 * (7 - M) + NV
370  IF Z$ <  > "Y" THEN 430
380 X =  PEEK (A0):Y = 0: IF X >
      128 THEN X = X - 128:Y = 128
390  FOR I = 0 TO 6:R(I) = X -  INT
      (X / 2) * 2
400 X =  INT (X / 2):Y = Y + R(I)
       * 2 ^ (6 - I)
410  NEXT I:A2 = A1 + 39 - NV - N
      V: POKE A2,Y
430  POKE A1, PEEK (A0)
440  NEXT N,M,Q,P
450  HOME : TEXT : INPUT "PLEASE
      NAMED FOR IT :";A$
460  PRINT D$"BSAVE "A$",A$4000,L
      $2000"
470  HOME : VTAB 10: HTAB 10: PRINT
      "***** GOODBYE!! *****": END
```

CP／M工具软件ED程序剖析（三）ED程序编辑缓冲区的结构和原理

中山大学　郭嵩山　王宗亮　曾超荣

一、编辑缓冲区的结构

ED程序把一个文件看作为一个长的字符串，为了编辑作业的方便，ED设置了一个假想字符指针（CP），编辑过程中，要根据（CP）的位置移动光标及发出编辑命令。

编辑作业绝大部分工作是在编辑缓冲区进行，编辑缓冲区类似于双向栈结构，它的首址为1D4D，其标号用BUF表示（本文标号是笔者自定的，下同），末址为1D4D十编辑缓冲区长度，其标号用BUFMAX表示，其中缓冲区长度要视CP／M系统容量而定（参看上期表1·1）。在结构上，编辑缓冲区将待编辑的文件以假想字符指针（CP）为界分成两部分，（CP）前的内容总是装到缓冲区的前半部分（低地址部分），（CP）后的内容总是装到缓冲区的后半部分（高地址部分），也即装到缓冲区末尾。除非缓冲区装满，否则前后缓冲区之间总是空的，我们将它称为空闲区。其结构见图3·1。

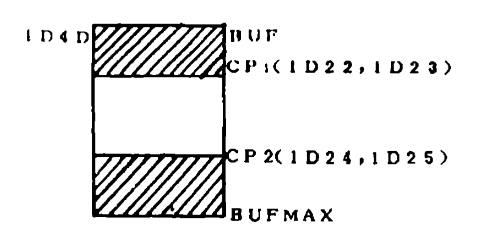


图3·1　编辑缓冲区的结构

前后两半缓冲区从逻辑上是连续的。因为（CP）仅仅是假想字符指针，并不象文件中某一个字符占据着实际位置。由于（CP）本身是可以移动的，所以前后两半缓冲区的大小也是浮动的。为了指示前后缓冲区的内容，ED专门设置了两个字符指针CP1（用1D22，1D23单元值来表示）和CP2（用1D24，1D25单元值来表示）。前半缓冲区从1D4D（BUF）开始，以CP1为界（不包括CP1），后半缓冲区从CP2＋1开始，到缓冲区末（BUFMAX）而止。也就是说，CP1、CP2代表假想字符指针CP，其中CP1用来指示缓冲区（CP）前的内容，CP2用来指示缓冲区（CP）后的内容，CP1和CP2间为编辑缓冲区的空闲空间。实质上，CP1、CP2就隐含了CP在文件中的位置。由于前后两半缓冲区是浮动的，所以CP1和CP2在缓冲区中的位置也是浮动的。在一般情况下，CP1和CP2在缓冲区中位置是不等的。这里有两种特殊情况：

1·**缓冲区空**　即进人编辑状态时，文件还未装入（或新编辑文件时未插入字符）。这时CP1与缓冲区首部（BUF）重合，CP2与缓冲区末尾（BUFMAX）重合。

2·**缓冲区满**　这时CP1和CP2重合，前后两半缓冲区在物理上也连续起来，整个缓冲区被编辑的内容所填满。

为了便于CP操作，ED程序专门另设了两个暂存字符指针的双字节单元，它们分别用标号BUFPT1（1D26，1D27）和BUFPT2（1D28，1D29）来表示。在一般情况下，它们分别用来指示CP1或CP2将要移到的位置。

二、子命令执行过程中编辑缓冲区字符指针的移动

ED程序在执行有关缓冲区操作的子命令时，其字符指针（CP1）（CP2）是如何移动呢？下面以插入字符（I命令），文本添加（A命令）、写行（W命令）、移字符指针到缓冲区开头和末尾（B命令）、移字符（C命令）、移行（L命令）、删字符（D命令）、删行（K命令）为例来说明。

1·**插入字符**

插入一个字符（执行I命令）到编辑缓冲区，字符指针（CP）就往后移一个字符位置，并总置

于最后插入的字符的后面。由于所插入字符总在（CP）之前，这时的字符，指针CP就用CP1来表示。插入字符到编辑缓冲区示意图如图3·2示出。图中括号部分表示插入命令前原来CP所在位置，斜线部分表示缓冲区已装载有内容。箭头表示CP移动的方向。

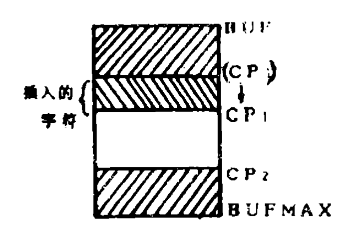


图3·2　插入字符到编辑缓冲区

这里有一种特殊情况，也就是当进行新文件编辑时，缓冲区空，CP1与缓冲区顶BUF重合，CP2与缓冲区底BUFMAX重合。这时执行插入命令，CP1在所插入字符之后，即CP1在输入文本末尾，CP2仍与BUFMAX重合。

2·添加文本到编辑缓冲区

添加文本到编辑缓冲区（即执行A命令时），所添加的文本是放在缓冲区中原有文本的后面。为此，ED程序先将（CP）移到缓冲区的末尾，然后将所添加的文本放到原有文件之后，文本添加完后，恢复CP到原来的位置，其示意图如图3·3所示，图中括号部分为CP原有位置。

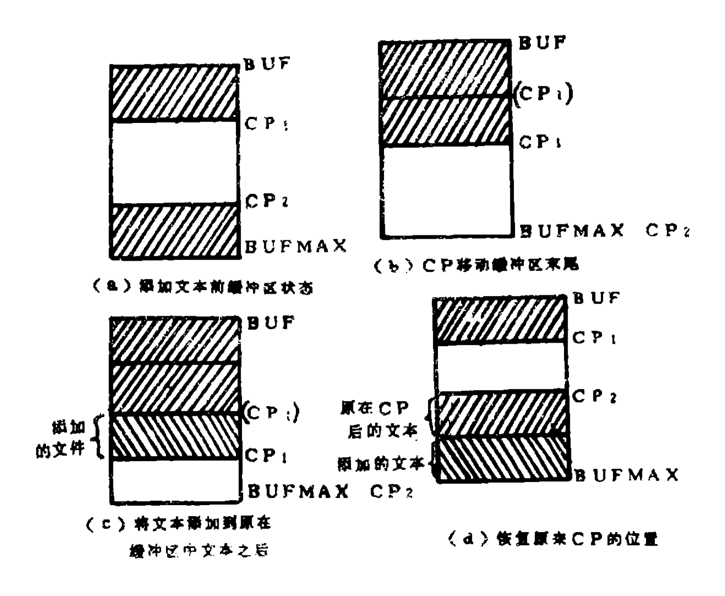


图3·3　添加文本到编辑缓冲区

这里有一种特殊情况，也就是在刚进入编辑状态时，此时编辑缓冲区空，CP1与BUF重合，CP2与BUFMAX重合。用A命令将文本填入编辑缓冲区后，（CP）位置不变，也即CP1仍与BUF重合，CP2和BUFMAX之间为所装入的文本。

3·写行到过渡文件

将编辑缓冲区前面n行写入到过渡文件（即执行W命令），ED程序先将CP移到缓冲区的开头，然后将n行写到过渡文件中。写完后，恢复CP原来的位置，其示意图如图3·4所示。图3·4中，写行时CP在文件末尾是随便设定的。

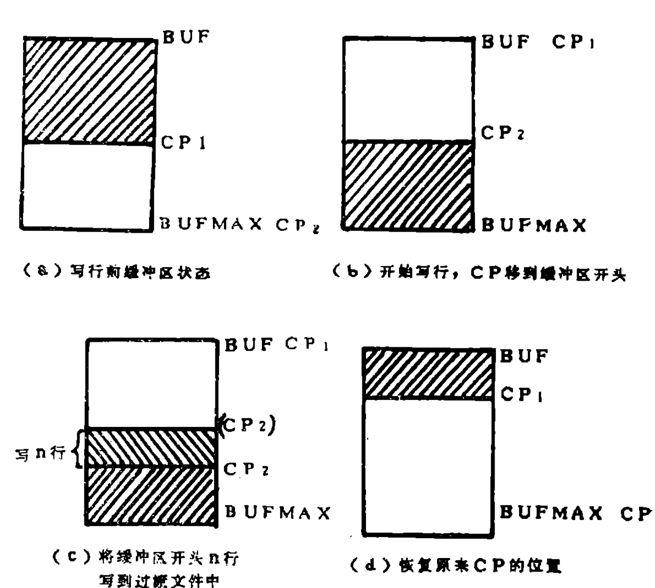


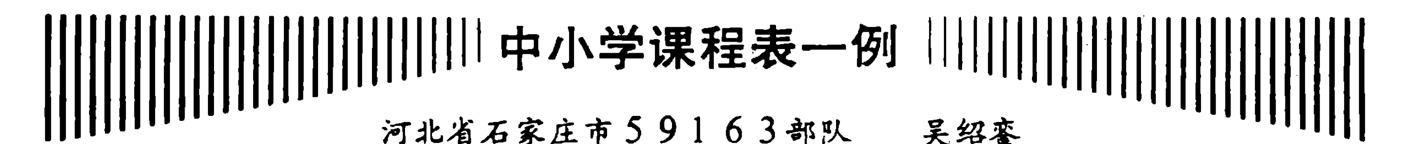
图3·4　从编辑缓冲区写n行到过渡文件

4·编辑缓冲区内当前字符指针CP的移动

ED程序的B、C、L命令都是在编辑缓冲区内移动CP的命令，这三条命令都有上移和下移两种情况：

1）·将CP上移，也就是执行B命令（CP移至缓冲区开头）、－nC命令（CP向前移n个字符）、－nL命令（CP向上移n行）。在执行中，把缓冲区中CP1往上移动所经过的字符顺次地传送到CP2开始往上的单元，在传送过程中，CP1、CP2也作相应的上移。

2）·将CP下移，也就是执行－B命令（CP移至缓冲区末尾）、nC命令（CP向后移n个



中小学课程表一例

河北省石家庄市59163部队　　吴绍銮

全国各地中学，有的已开展计算机普及教育，吸引了广大青少年喜爱，积极参加了课余计算机的学习小组，并取得了可喜的成绩。为了他(她)们学习得更好，结合学生的实际应用，我用BASIC语言，设计一条中小学生课程表程序一例。在IBM—PC／XT机通过。

一、操作简单，对好打印纸后，按RUN键[回车]，立即打印出一张课程表。

二、课程的内容，根据学校的安排，在DATA语句中输入或修改。

下面是程序及运行结果：

```
5 REM IM .BAS 课程表程序设计：吴绍銮
10 DIM P$(66)
15 WIDTH "LPT1:",80
20 P$="————————————————————————————————————————————————————————————"
35 D$="|——————————+————————+————————+————————+————————+————————+——————|"
40 LPRINT CHR$(27);"I";"C"
45 LPRINT TAB(24);"高 一 课 程 表"
```

字符)、nL命令(CP向下移n行)，则把缓冲区中CP2往下移动所经过的字符顺次地传送到CP1开始往下的单元。传送过程中，CP1、CP2也作相应的下移。

无论是CP1、CP2的上移或下移，总是保持CP1、CP2之间区域为空闲区，这与前面提到的编辑缓冲区的结构的定义是一致的。

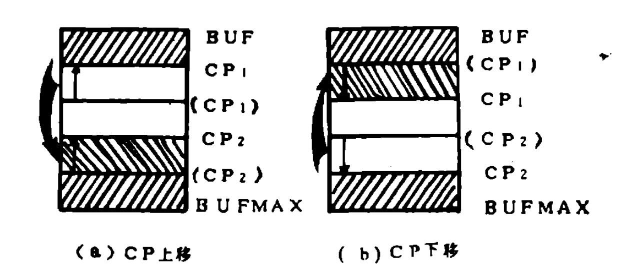


图3·5　当前字符指针CP的移动

图3·5示出了编辑缓冲区当前字符指针CP移动的示意图。图中括号部分表示CP原来的位置，箭头表示CP移动的方向。

5·删除字符操作

ED程序的D、K命令是删除CP前后字符的命令。这里也有删CP前字符和删CP后字符两种情况。

1)·删CP前字符，也就是执行－nD命令(删CP前n个字符)、－nK命令(删CP前n行字符)。执行命令时，根据n值，将CP向前移n个字符(－nD时)或向前移n行(－nK时)，将CP1所经过单元变成空闲区。

2)·删CP后字符，也就是执行nD命令(删CP后n个字符)、nK命令(删CP后n行字符)，执行命令时，根据n值，将CP向后移n个字符(nD时)或向后移n行(nK时)，将CP2所经过的区域变成空闲区。

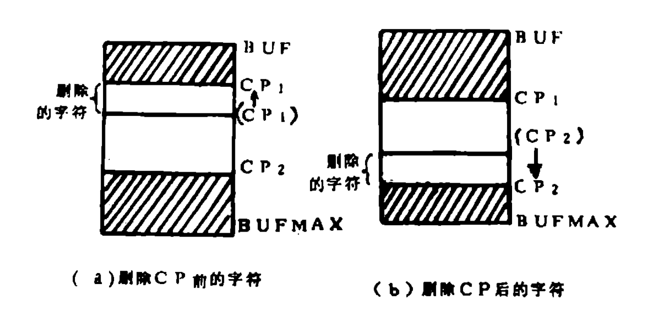


图3·6　删除字符操作

图3·6示出了编辑缓冲区中删除字符操作的示意图，图中括号部分表示CP原来位置，箭头表示CP移动的方向。删除字符操作实际上是在删除字符时，向前删只移动CP1，向后删只移动CP2，而不象移字符(或移行)命令那样，把CP所经过的字符送到对应缓冲区中，这样来实现删字符的目的。

```
50 LPRINT CHR$(27);"I";"B"
55 LPRINT TAB(23);"~~~~~~~~~~~~~~~~~~~~~~~~"
60 LPRINT TAB(3);"石市铁一中"TAB(38);"          1986年 8 月31日"
65 LPRINT P$
70 FOR I=1 TO 5
75 READ X,X$,X1$,X2$,X3$,X4$,X5$,X6$
80 IF I>5 THEN 115
90 LPRINT TAB(1);"|";X;TAB(5);"|";X$;TAB(11);"|";
X1$;TAB(19);"|";X2$;TAB(27);"|";X3$;TAB(35);"|";X4$;
100 LPRINT TAB(43);"|";X5$;TAB(51);"|";X6$;TAB(59);
"|";
105 LPRINT TAB(1);D$
110 NEXT I
115 LPRINT "  "
120 LPRINT "  "
125 LPRINT TAB(1);P$
130 FOR L=1 TO 4
132 READ X,X$,X1$,X2$,X3$,X4$,X5$,X6$
145 LPRINT TAB(1);"|";X;TAB(5);"|";X$;TAB(11);"|";
X1$;TAB(19);"|";X2$;TAB(27);"|";X3$;TAB(35);"|";X4$;
150 LPRINT TAB(43);"|";X5$;TAB(51);"|";X6$;TAB(59);
"|";
155 LPRINT TAB(1);D$
160 NEXT L
165 END
200 DATA 0,"   ","星期一","星期二","星期三","星期四","星期五","星期六"
210 DATA 1,"8:25","历史","外语","语文","物理","代数","外语"
220 DATA 2,"9:20","代数","几何","物理","化学","物理","语文"
230 DATA 3,"10:15","语文","语文","代数","几何","历史","语文"
240 DATA 4,"11:10","化学","物理","外语","地理","外语","几何"
250 DATA 5,"3:00","体育","政治","地理","政治","体育","化学"
260 DATA 6,"0:00","  ","  ","  ","  ","  ","  "
270 DATA 7,"0:00","  ","  ","  ","  ","  ","  "
280 DATA 8,"0:00","  ","  ","  ","  ","  ","  "
```

高一课程表

~~~~~~~~~~~~~~~~~~~~~~~~

石市铁一中　　　　　　　　1986年 8 月31日

0		星期一	星期二	星期三	星期四	星期五	星期六
1	8:25	历史	外语	语文	物理	代数	外语
2	9:20	代数	几何	物理	化学	物理	语文
3	10:15	语文	语文	代数	几何	历史	语文
4	11:10	化学	物理	外语	地理	外语	几何

5	13:00	体育	政治	地理	政治	体育	化学
6	10:00						
7	10:00						
8	10:00						
~~~~~~~~~~~~~~~~~~~~~~~~

增大APPLE Ⅱ仓颉汉字系统程序区的方法

张美笑

在用APPLE Ⅱ仓颉汉字系统（C－PLUS Ⅱ或HZK－Ⅰ）来处理象图书馆业务管理方面的问题，往往会因程序区不足而烦恼。但是，如果我们利用修改指针，进行程序的驳接和复盖技术，就可以满意地解决，使程序设计合理，内存充分利用。程序盖技术是大家熟知的，本文仅对修改指针，和驳接程序两种扩大程序区方法加以叙述。

一．仓颉汉字系统内存用户区

在APPLE Ⅱ机的I／O槽插入一片汉卡，开机后键入命令PR#CN（CN为汉卡所在的I／O槽号），系统就转入汉字系统。如果要运行一个处理中文信息的BASIC程序，还需要先执行一段中文自启动程序方可。当系统转入汉字系统状态后，就要占用两个内存区：一个是从地址4000H到5FFFH共8KB，作为汉字显示RAM区；另一个是地址从9000H到95FFH，作为汉字组字的RAM区。后者由中文自启动程序中的HIMEM：36864去拨动内存高位指针，使其建立一个保护区，归系统专用；前者将内存用户区拦腰折断成两半：前半从0800H到3FFFH共14KB，后半从6000H到8FFFH，共12KB。由于前、后两个用户区不连续，在键入程序超过14KB后，系统无法接受超过14KB后的部分。诚然，尽管你可以键入一个比14KB稍小的程序，但也不一定能正常运行，往往在程序运行不一会，屏幕上出现许多雪花，无法继续运行下去。

二．修改LOMEM内存低限指针，使系统能运行稍小于14KB的程序

为什么汉字系统不能正常运行稍小于或等于14KB的程序呢？这是因为每个程序都有变量，有些还有数组，它们随着程序运行逐一产生，并尾随程序后存放在内存里的。一旦变量、数组区增大并落入4000H～5FFFH区域时，机器就会出现上述故障现象，程序无法继续执行。

接近14KB长度的程序是经常可能存在的，解决这个问题的方法，就是将变量、数组存放到6000H单元以后的另一个用户区，而不要尾随在程序的后面内存空间。这种想法，可通过改变指针来达成，即：

在程序开始处，执行一句LOMEM：24576语句，用以改变内存低限指针，指到6000H下半用户区的开始位置。当执行该语句后，变量开始指针（69H，6AH单元值）和数组变量开始指针（6BH，6CH单元值）自动改变。当程序运行而产生的变量和数组，不会尾随程序后部存放，而是存放到6000H单元以后的内存区。

这一方法简便有效，当您发现自编的程序运行不一会屏幕出现雪花，无法运行时，只要在程序开始处加一条LOMEM：24576语句便可以得到解决。

三．用程序的接驳方法解决超过14KB长的程序

在编写程序时，往往会超过14KB长，但又超过不多，是否可以不必用程序复盖的办法来解决呢？这是我们在编制图书馆中文期刊订购检索系统时所遇到的，上述已经提到仓颉汉字系统有两个用户区，我们能否将程序放入这两个区内呢？即程序占满了前半14KB区后（通常应稍稍小于14KB），跨越中文显示区，放入到后半12KB区域。如果能够实现，就可以增大仓颉汉字系统的程序区了。当然这种增大是有一定限度的，因为总的用户区只有26KB所限，还加上需考虑变量、数组和字符串要占用一部分内存空间。尽管如此，当程序稍比14KB大，而又不易将它分成两个独立的短程序时，上述的想法是一种解决问题的好方法，它既可以加快程序运行速度，又可减轻编程的工作。

要将一个程序置于两个不连续的用户区的方法，就是下面要讲述的程序接驳法。接驳法是基于BASIC程序在内存里按链结构方式贮存，只要人为改变其链，就可以变更BASIC程序的结构。下面以例说明这个原理。

1．一个具有五行语句的BASIC程序

```
5 POKE 2072，62
10 PRINT "AA"
20 PRINT "BB"
```

30 REM DEMONSTRATION PROGRAM

40 PRINT "AABB"

2. 键入CALL －151，转入监控后查看0800H至085FH内容

```
]CALL -151

*0800.085F

0800- 00 0E 08 05 00 B9 32 30
0808- 37 32 2C 36 32 00 18 08
0810- 0A 00 BA 22 41 41 22 00
0818- 22 08 14 00 BA 22 42 42
0820- 22 00 3E 08 1E 00 B2 20
0828- 44 45 4D 4F 4E 53 54 52
0830- 41 54 49 4F 4E 20 50 52
0838- 4F 47 52 41 4D 00 4A 08
0840- 28 00 BA 22 41 41 42 42
0848- 22 00 00 00 FF FF FF FF
0850- FF FF FF FF FF FF FF FF
0858- FF FF FF FF FF FF FF FF
```

3. 从中可以看出BASIC的链结构如下：

程序从0801H单元开始存放，0801H，0802H单元为链，其值是080EH，指出下一行语句开始存放的单元地址；0803H，0804H单元为行号（0005H行），其后是行5语句的内容，直至080D单元值为00表示该语句行结束。自080EH单元开始到0817H单元存放行10语句，如此重复，直到084AH，084B链单元值为00，表示程序结束。

4. 将第3行语句（行号为20）的链改成083EH，即只要将2072号单元的34值（22H）改为62（3EH）即可。原程序行5 POKE2072，62就是用于此目的，一旦程序运行，链被改变，语句30就自动消失。

根据上述原理，我们可以按如下方法去接驳程序，使一个大于14KB程序分别置入前、后半内存的用户区：

1. 将一个超过14KB的程序，机械地分为两部分，前部分的首行语句予先写成：

POKE ×××××，××：POKE ×××××，××

语句中的每个"×"表示任一个十进制数字，待最后修正。该部分程序段长度取其接近14KB（稍小于14KB）。

2. 前后两部分的语句行号应相隔一个较大的数。并分别以文件的形式存入磁盘，假定前部分文件名为F1，后部分文件名为F2。

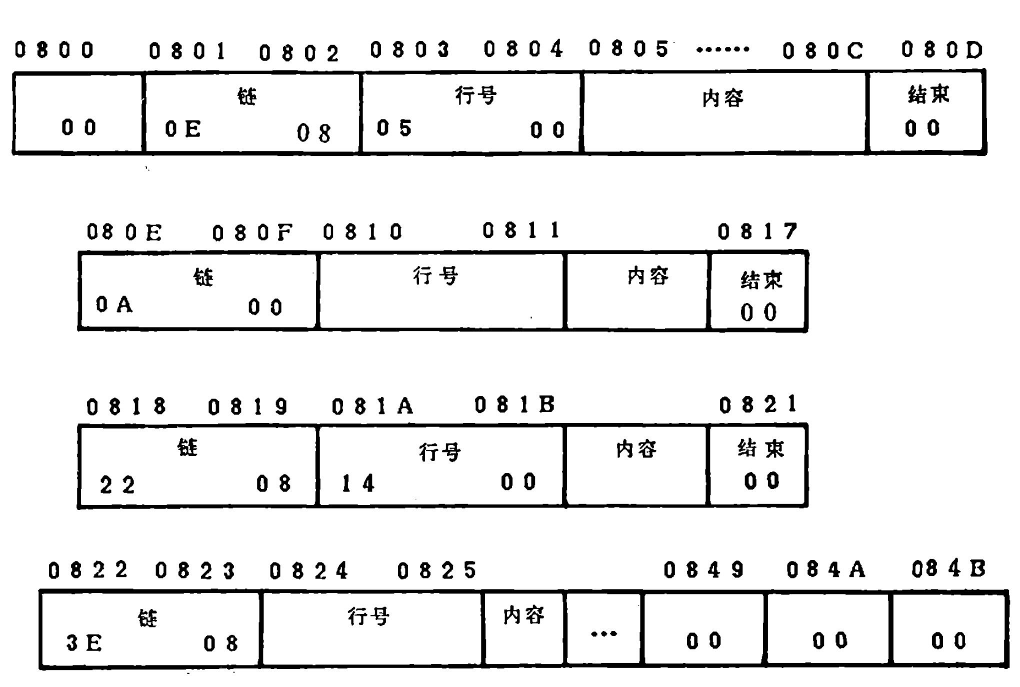
3. 随意建立一个非执行的无用的稍大于8KB长度的程序，语句行号取值为程序F1的最大行号与程序F2的最小行号之间，并以文件名F3存入磁盘。

4. 在英文状态下，用RENUMBER程序将F1、F3和F1合并成一个大程序。其步骤：

（1）将系统盘中的RENUMBER复制到存放F1，F2，F3的磁盘上；

（2）RUN RENUMBER<CR>将RENUMBER装入内存高位区，并运行；

（3）打入LOAD F1<CR>，将合并的第1个程序调入内存；





蛇吃豆

北京市第一〇五中学高二(四)班　赵效东

该程序比较有趣。程序运行时，显示器四周画上一个框，机器随机地在框中出现一颗“豆”，框的左上角爬进一条“蛇”，游戏者用↑　↓　←　→控制蛇的运动，蛇每吃一粒豆便身长增加5个单位。游戏规则是蛇不能碰框和自已的身体，这样每吃一粒豆就增加了一定难度，当吃了第40个豆时，程序便显示“WON!!　AGAIN?(Y/N):”这时按“Y”则重新开始。如果在这过程中蛇碰了框或自已身体，则显示“LOSE!!　AGAIN?(Y/N):”这时按“Y”则重新开始。

本程序有一缺点就是没有音响，该程序适用机型：SHARP　MZ731，语言BASIC。

```
10 CLS:DIM X(220),Y(220),A(40,79)
20 FOR I=1 TO 79:SET I,0:A(0,I)=1:SET I,
40:A(40,I)=1:NEXT
30 FOR J=0 TO 40:SET 0,J:A(J,0)=1:SET 79
,J:A(J,79)=1:NEXT
40 U=10:T=0:JY=1:JX=0:X=1:Y=U+1:M=0
50 FOR I=1 TO U+1:SET 1,I:X(I-1)=1:Y(I-1
)=I:A(I,1)=1:NEXT
60 RX=INT(RND(1)*78+1):RY=INT(RND(1)*39+
1): IF A(RY,RX)=1 THEN 60
70 SET RX,RY
80 GET K$:CURSOR 2,22:PRINT "MARK: ";M
90 IF(K$="↑")*(JY=0) THEN JX=0:JY=-1
100 IF(K$="↓")*(JY=0) THEN JX=0:JY=1
110 IF(K$="←")*(JX=0) THEN JY=0:JX=-1
120 IF(K$="→")*(JX=0) THEN JY=0:JX=1
130 X=X+JX:Y=Y+JY
140 IF A(Y,X)=1 THEN 180
150 SET X,Y:A(Y,X)=1:RESET X(T),Y(T):A(Y
(T),X(T))=0:X(T)=X:Y(T)=Y:T=T+1:IF T>U T
HEN T=0
160 IF (X=RX)*(Y=RY) THEN U=U+5:M=M+1:GO
SUB220:GOTO 60
170 GOTO 80
180 CURSOR 0,23:PRINT "LOSE!!        AGAI
N?(Y/N):";
190 GET K$:IF K$="Y" THEN RUN
200 IF K$<>"N" THEN 190
210 END
220 FOR I=0 TO 4:X(U-I)=X(T):Y(U-I)=Y(T)
:NEXT:IF M<40 THEN RETURN
230 CURSOR 0,23:PRINT "WON!!         AGAI
N?(Y/N):";
240 GET K$:IF K$="Y" THEN RUN
```

---

(4)打入&H<CR>，第1个程序便会处在HOLD状态；

(5)打入LOAD　F3<CR>，以便将合并的第2个程序装入；

(6)打入&M<CR>，将F1和F3合并。

(7)用SAVE文件名的命令将合并后的程序存入磁盘。

重复上述的操作，就可以将F1，F3，F2合并成一个大程序，并且使F1占据内存前半用户区，F3占据中文显示RAM区，F2占据内存后半用户区。

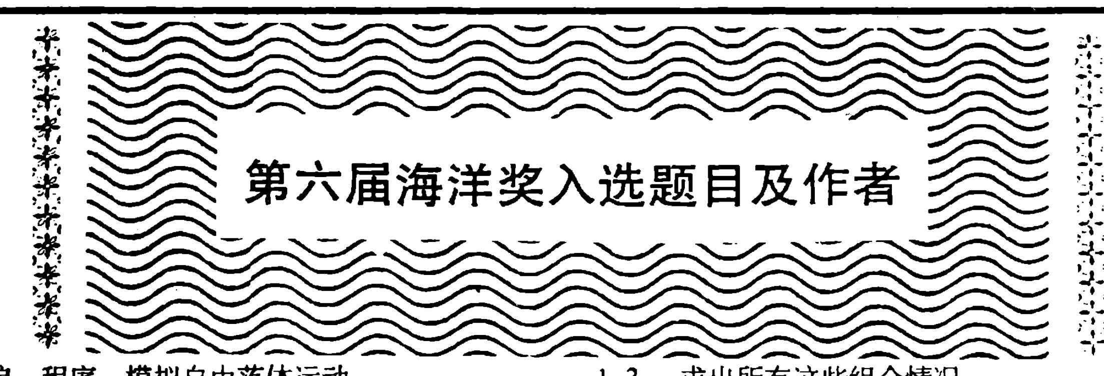
5·在英文状态下调入合并后的大程序(令其文件名为MYFILE)，转入监控状态，查看原先F1程序最后一行语句的链单元地址(即最后一行语句开始所在内存的单元地址，共两个)。记下第一个单元地址(AL1)，第二个单元地址(AH1)；再查看原先F2程序在内存的首址。并记下高8位值(AH2)，低8位值(AL2)。(注AH1，AL1，AH2，AL2均为十进制数值)。

6·将系统转回APPLESOFT状态，修改程序首行　POKE　×××××，××：POKE　×××××，××语句为：

POKE　AL1，AL2：　POKE　AH1，AH2

然后，用SAVE　MYFILE命令存入磁盘。

这样合并修改后的MYFILE就是一个能使F1，F2文件连在一起并分别置于内存用户区的前半部分，和后半部分。该程序(MYFILE)必须在英文状态下调入并运行，一旦执行POKE　AL1，AL2：　POKE　AH1，AH2语句后，就会改变程序的链，程序中的原F1部分执行完毕后，自动地跳过无效部分(F3)，直接链接到原F2部分。由于使用一个无用的部分去填充了中文显示RAM区，所以在程序转入中文状态后，无丝毫的影响，这样，一个长度超过14KB处理中文的程序就能在汉字系统状态下正确运行。被中文显示RAM区所折断的用户区就好象变成一个能连续存贮和运行程序的用户区，在使用者看来，程序区的空间已增大了。



第六届海洋奖入选题目及作者

(1)编一程序，模拟自由落体运动。

（吉林省长春市三十七中学八四(一) 潘 锋）

(2) 有一正整数，与它自己相加、相减、相乘、相除，把所得的和、差、积、商四个数加起来，恰好等于64，编一BASIC程序，找出此数。

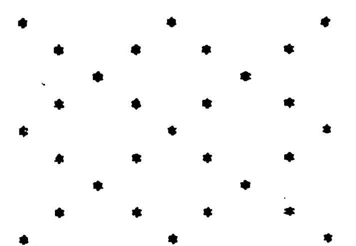
（天津广播电台 孟玉贞）

（湖北省武汉市武昌县纸坊一小 严祥贞）

(3)求出等式1/a+1/b+1/c=n/(n+1)的所有正数解，其中a，b，c，n都是自然数。

（山东省东营部华东石油学院附中高一(一)班 初 鹰）

(4)编一程序显示下面图形，并使图形可扩大缩小。



（江西省龙南中学 刘江阳）

(5)小刚的爸爸用一个可装8斤油的瓶从集市上买8斤油回家，想分出4斤油给小刚的姐姐，可是找遍所有的地方，只找到能装3斤油和5斤油的瓶子各一个，这时小刚灵机一动，想出个妙办法，顺利地分出4斤油，请问小刚是如何干的？（注意：使你的程序保证倒油次数为最小？）

（浙江省鄞县中学 毛宇明）

(6)现有1000人投票选厂长，候选人编号为A、B、C、D、E、G，编一统计选票的程序。

（江西省新平县683号信箱 邓 潮）

(7)编一程序，使输入的6502汇编语言以Z80汇编语言的形式打印出来。

（惠州市第七中学高三(3) 周玉光）

(8)已知1，2，…，13这十三个数，试从中取出N（1<N<4）个数组成一组，使其和等于13，求出所有这些组合情况。

（武汉水运工程学院275信箱 陈朝晖）

(9)试编一个程序，在屏幕上打印一个走动的“$”。要求“$”从第一行第一列开始，走完第一行后，从第二行尾返回，再从第三行开始走完第三行后又从四行末返回，如此下去一直到“$”走完一屏。

（华南师范大学物理系八五(3) 姚志良）

(10)编一程序，演示我们在纸上做乘法运算的全过程。两乘数为正整数，并且最后一个非零数字之前的数位不超过8位。例子：

```
   1040
 × 2700
 -------
    728
  208
 -------
 2808000
```

（南雄县南雄中学高二(3) 朱永家）

以上每道题目作者均奖励海洋牌双面双密度电脑软磁盘一片（奖品由深圳市南头较场后海洋音响器材总公司深圳分公司电脑磁盘厂资助。该厂生产电脑软磁盘、录音带、原声带、空白磁带等。电话：23711转45342，电挂：0742）。

本刊87年7期将公布第六届题目征答人选者名单及答案选登，87年4期将公布第七届海洋奖入选题目，第5期公布第四届征答人选作者名单及征答选，欢迎踊跃参加征题征答活动，寄来的题目答案均人选者奖励电脑软磁盘2片。来稿请于信封上面写上“征题征答”字样。

本刊连续12届开展有奖征题征答，每届奖励20名，征题征答对象均不限。

寄来的题目（要求自编）请提供参考程序。程序及运行结果请打印在同一张纸上。

每届全部题目都答或选答均可。每道程序及运行结果均占一张纸，每张纸上面均写“×届×题答案”，纸下方写上作者姓名及详细地址。

M24显示控制板的修复

国营杭州磁记录设备厂计算机室　朱其良　周刚尧

M24是意大利Olivetti公司生产的IBM PC兼容机，已由云南电子设备厂等单位引进开发生产，其主机CPU采用8086，通过总线转接板与外部连接的数据总线可为8位或16位，较IBM PC有速度快，扩展槽数目多（七个槽口）等特点，在国内拥有众多的用户。由于主机箱结构紧凑，显示控制板，软盘控制板不是直接插在扩展槽上，从而给维修带来不便，现就显示控制板出现的一故障，检测及排除方法作一介绍。

一．故障现象

显示器工作在字母数字方式下，开始屏幕显示正常，而当屏幕显示满一屏时，屏幕的显示发生变化，原来没有字符的地方出现"$"字符，而原来所显示的字符则变成另外字符。此时，磁盘的读写及打印机工作均正常。

二．故障分析及检测

从故障现象看，其它外设工作均正常，可初步认为主机方面基本上没有毛病（若有毛病的话，其它外设也会有所表现），这样即可初步判断问题出在显示控制板上。

显示控制器是M24系统和显示器之间的一个接口，它支持单色或彩色的显示器，它有两种基本的操作模式：一是字母数字混合编制（简称为A／N）二是全指针寻址图形模式（简称为APA），A／N模式包括：硬件光标和视频接收、闪烁、醒目、隐蔽及底线等可见属性。

显示控制器和M24系统采用两种通信方式：一，输入／输出方式，M24发送各种命令到显示控制器，控制某些方面的显示，如光标位置，光标大小，清除和重置显示及改变操作方式。二，存贮器映象，计算机写信息到控制器中的存贮器，并且屏幕连续地扫描存贮器，把存贮器中的数据在屏幕显示出来，CPU通过读存贮器中的内容控制显示。

显示控制器有一块32K的RAM，其地址为B8000～BFFFF，在这RAM中有一个判断电路使CPU和显示器在没有更多干扰的情况下能够极快地使用显示存贮器，25×80的字母数字方式电路没有双重的通道，因而更新只发生在水平或垂直回扫期间。

控制器工作在字母数字方式时，屏幕上每一个显示位置，占用存贮器的两个字节（即每16位定义一个字符），第一个字节（偶数地址）规定有待在屏幕上显示的字符ASCII码。第二个字节（奇数地址）定义为属性字节，如下表：

	属性字节	ASCII代码字节
地址	B8001	B8000
位	15 14 11 12 10 9 8	7 6 5 4 3 2 1 0

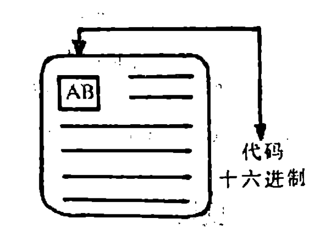
在彩色显示情况下，它作为控制颜色用，而在单色显示情况下，它作为控制色调的灰度，在上述两种显示形式中，它还控制闪烁，亮度和底线。

在屏幕左上角第一个位置占用显示存贮器初始的两个字节，下一个位置，即右边第一列，占用存贮器其次的两个存贮器单元，如此一直继续到第一行的结尾（共用了160个存贮器单元），在每一行结尾，存贮器映象立刻转到下行的开始，以致下一行的第一列被映象成为前一行最后列下面的一对字节（即从第161，162存贮单元开始），显示存贮器的使用是不间断的，上一行结束下一行就开始，每显示一页共需4000个存贮单元。见图(1)所示。

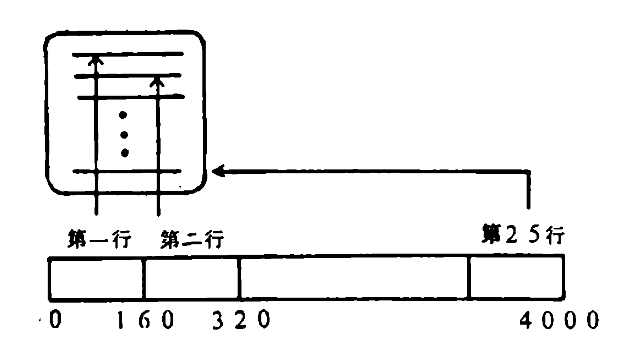
根据上述的显示控制器工作原理，进一步分析显示故障，发现显示出错按照一定的规律：字符所对应的ASCII码的第2位总是置"1"，如0、1、2显示成4、5、6，A、B、C成为E、F、G，而显示4、5、6、E、F、G等字符则不出错。

针对上述情况，
运行测试程序Customer：

A＞ Customer ［回车］



A	属性字节	B	属性字节	···
A1		A2		···



图(1)显示存贮器映象

```
   * * * Customer Test program * * *
       Version 1.05 July 20,1984
=⇒Press "Ctrl Break" keys to exit test
program
  Do you wish to rum the tests in au
tamatie mode?
                ("0"=No,    "1"=Yes)
按0按回车
  Testing Options
  0-Execute Tests once
  1-Format scratch Qiskette
  2-Exit Diagnositics Program
  Enter Desired Action
按0按回车
  The Installed Hardware is:
  1-Main Board
  3-256kB Memory
  4-Dater Keyboard
  5-Monochrome Video System
  6-1 Parallel port
  7-1 Serial port (using 8250)
  8-1 48 TPI Floppy Disk drive(s)
  10- Parallel printer (LPT1:)
  Is the list of hardware correc
t?
              ("0"=NO,    "1"=YES)
按1按回车
  1-Main Board
  3-256kB Memory
  4-83Key keyboard
  5-Monochrome Video System
  6-1 Parellel Port
  7-1 Serial Port (using 8250)
  8-1 48TPI Floppy Disk Drive(s)
  10- Parallel Printer (LPT1:)
  To test all listed hardware pre
ss 'Enter' key only
  To test indiridual hardware fun
ctions, Enter the clesived test
number(s)
  With spaces between the numbers,
and then press the 'Enter' key
按5按回车
  Pause on error ('0'=No,  '1'=Yes)
按1按回车
```

等待数秒钟后，显示：

```
 501 Error
     Error Monochrome Vided RAM
       RAM  Segment  offest    5ASA
525E
       16384  B800    7FFE    Writter
Read
```

501出错为视频RAM出错，共出错16K字节段地址和偏移量为B800：7FFE。当主机CPU向显示控制器的RAM写码字“5A5A”读出时为“525E”即从写入的“01011010 01011010”到读出的“01010010 01011110”，码字的第2位和第11位出现了错误，第2位冒“1”，第11位丢“1”。

由显示控制器工作原理知道：在字符数字状态时，偶数字节为ASCII代码，奇数字节为属性字节。可见，在显示字符和显示属性这两方面都有故障存在。

从测试程序的结果来看，RAM在读写时全错。从存贮器存取的工作原理(这部分原理从略)来看，要使RAM全部出错，这种可能性比较小，而由数据总线向显示器RAM写数据或数据总线从RAM读数据时，显示器控制板上的数据总线缓冲电路中的某些集成片子失效引起的可能性比较大。

如图(2)所示，从CPU到显示器缓冲器的数据D0～D15首先通过双向数据总线缓冲，这个双向数据总线缓冲是由两块74LS373缓冲器和两块74LS244缓冲器所组成，数据总线缓冲的方向或者由数据流向CPU的缓冲器LS373的使能端，或者由数据流向显示缓冲器LS244缓冲的使能端来决定的。当BOARDW信号变低时两个LS244便使能工作，然后数据位D0～D7传送到IND0～IND7上，D8～D15传送到IND8～IND15上(见图(2)所示)。当-BOARDRDL和-BOARDRDU信号变为低时两块LS373缓冲器使能。而当信号-BUSRDL变高时，为CRTC使用的独立存贮器的数据被锁存入两块LS373缓冲器，于是数据位IND0～IND7传送到D0～D7，IND8～IND15传送到D8～D15上。

从CPU来的数据通过数据总线缓冲以后传送到CPU数据缓冲器是由两块三态收发的LS245组成的。数据缓冲器LS245被地址译码电流信号-RAMBUF变为低时使能工作。无论什么时候CPU总对RAM缓冲存取，这一数据缓冲器的方向由信号-BOARDW决定。当它为低时，方向流向RAM缓冲；为高时，方向为流向CPU。

CRT数据缓冲接收从RAM缓冲来的数据D

RAMDAT0 ～DRAMDAT1 5，缓冲锁存总是使能的，只有在时钟信号转变时，DRAMDAT0 ～DRAMDAT1 5 才被锁存到输出端。

在数据总线缓冲电路中，这些片子对RAM的读写有着至关重要的作用，若因该电路中的74LS244，74LS373或74LS245所对应的D0 ～D1 5 中的一位失效，即该位总是置“1 ”或置“0 ”，就有可能在读写RAM时出现全部出错的现象。因此，必须把重点放在检查缓冲电路上。

为了进一步缩小检查范围，以区别是读屏幕(读存贮器）错，还是写屏幕(写存贮器）错，在DEBUG管理下利用INT10H，我们编写了几条指令检查了一下读写RAM。检查结果，发现读屏幕时数据第2 位冒“1 ”。这从硬件上看，读数据时通过LS244和LS245。由于D8 ～D1 5 是属性字，无法检查到，但故障很可能就在LS373，LS244，LS245中。

三．故障排除方法：

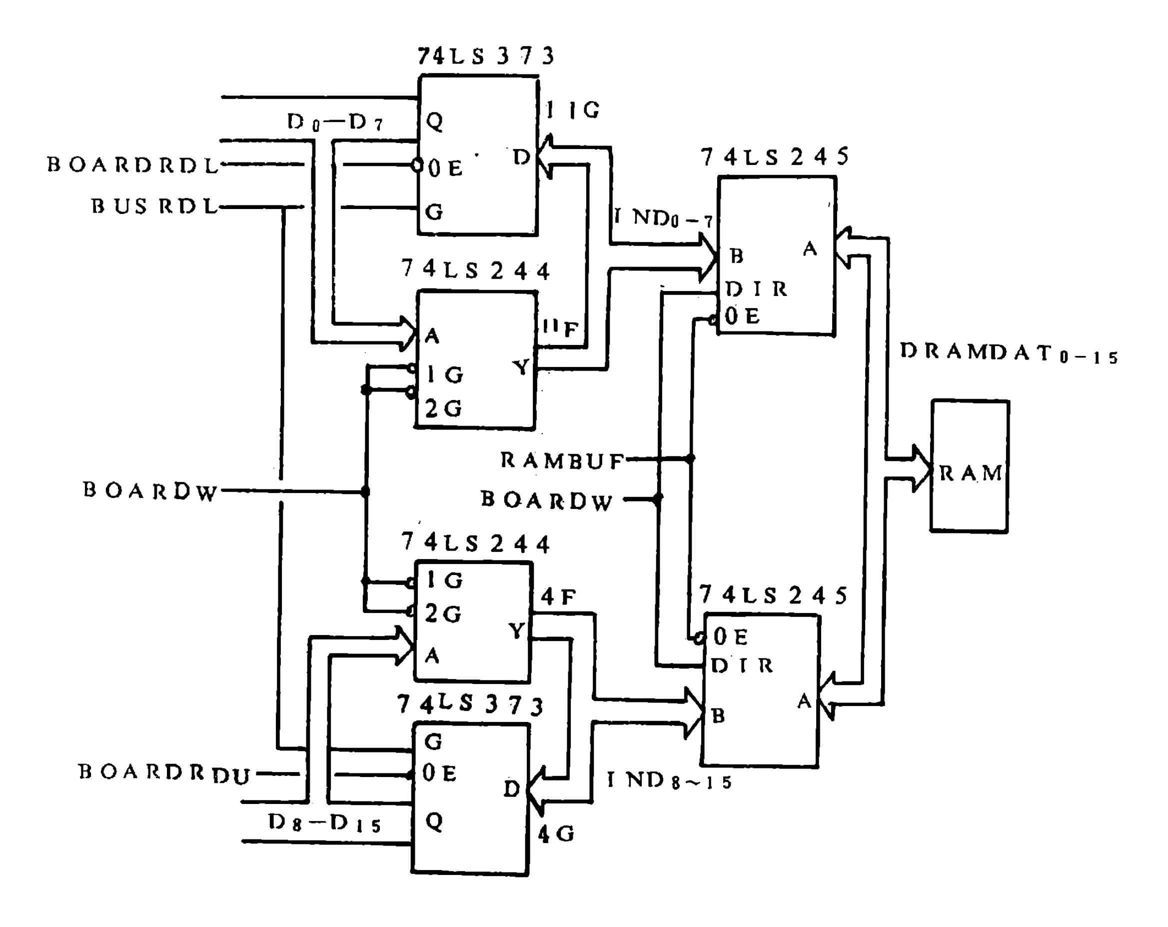
在没有完全查出故障源之前，为了不轻易取下片子并能正确地查出片子故障，采用局部检查法：

1．认清数据出错的第2 位和第1 1 位所对应的片子脚号，把相同型号的好片子连接上工作电压以及有关的使能信号

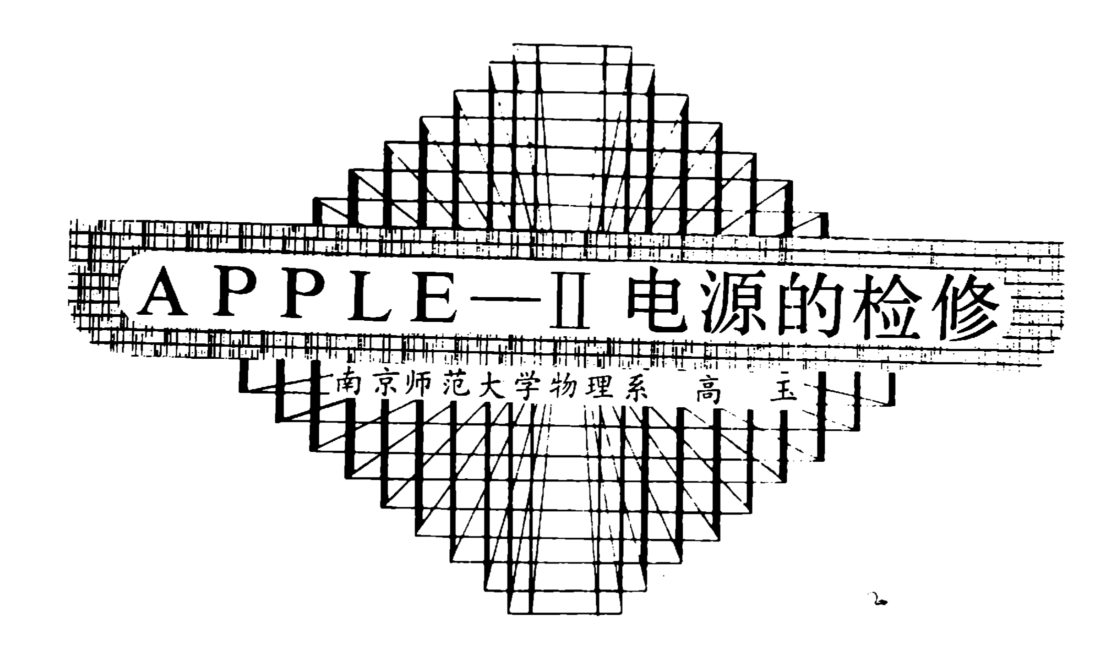
2．剪断所怀疑片子的相应的脚，并使之脱开原电路，把电路连接到好的片子上，然后开机观察故障是否消失。若没有消失，先取下好片子，恢复原来的片子，然后再换另外几块有关的片子作同样的检查。这样共做了四、五次，就查到了故障片子：74LS244(4F)和74LS245(9F)

3．焊下这两块片子，焊上同型号的好片子，故障即消失。

小结：74LS245所对应的数据第2 位失效老是冒“1 ”，因该位失效，引起数据错，74LS244所对应的数据第1 1 位失效，总是置“0 ”，引起显示属性错。



图(2)数据总线缓冲电路



APPLE—Ⅱ电源的检修

南京师范大学物理系 高 玉

APPLE－Ⅱ电源是较易发生故障的部件。该电源采用脉冲式电源，在检修过程中发现其线路因各机型而异，与原苹果机资料中提供的线路有较多的变动。为便于读者检修，这里提供一个根据实际电源而画出的较为详细的原理图。

如图，220V交流电压经高频滤波器后直接桥式整流，经电容滤波后得到直流电压。(约300V)此电压加至一高反压大功率晶体管Q_1作脉冲振荡，高频交流电压经一小型变压器T耦合到次级，经过整流二极管和滤波电容器可得到±5V，±12V四组直流电压输出。

电路中L_1为脉冲振荡主绕组，L_2和D_3为反向高压脉冲提供通路，L_3为反馈绕组，L_3一端接地，另一端通过R_8，C_3接至Q_1的基极，形成正反馈。Q_1的偏置电路由D_1和R_5组成，R_5一端接Q_1的基极，另一端接由R_2R_3组成的分压器上。这样由Q_1，L_1，L_3，R_5，D_1组成了脉冲电源的自激振荡回路。

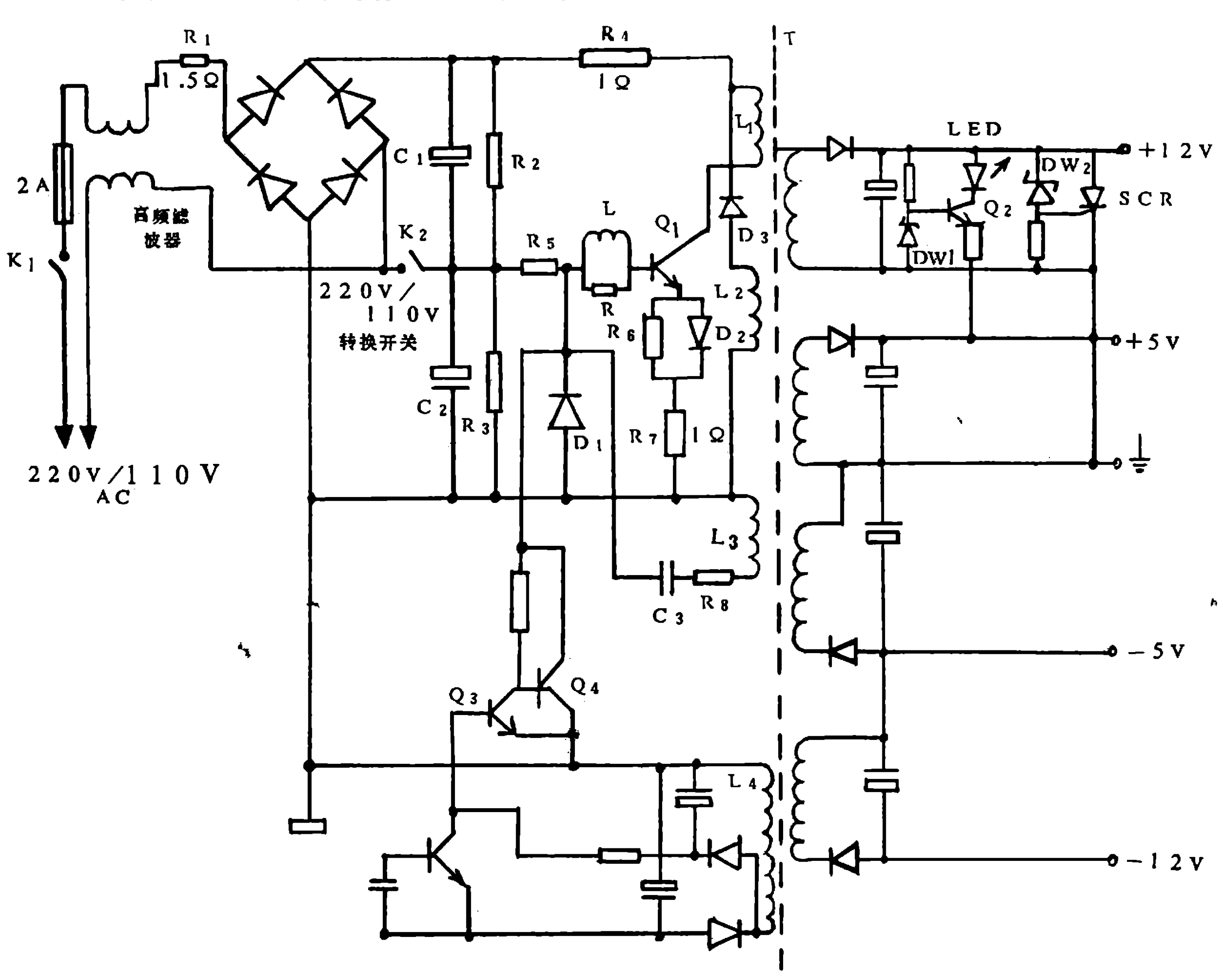
在Q_1的基极还接有由Q_3，Q_4，光耦合器，Q_2等组成的控制电路，其中Q_3，Q_4，光耦合器的接受端由L_4绕组通过整流后提供直流电源。控制电路的作用是稳定输出电压，其工作原理如下：

控制电路中的Q_2接在电压输出端，实际上是一个比较器，其基极接DW_1上，DW_1从+12V上取得一稳定电压，Q_2的发射极接+5V电位上，Q_2的集电极接光耦合器的发送端(发光管LED)。如果+5V电压下降，Q_2的发射极电位下降，而Q_2的基极电位稳定，这就造成Q_2集电极电流增大，使得发光二极管LED发出更亮的光，光耦合到光耦合器的接受端，增加光电三极管的导通量，其变化通过Q_3Q_4组成的直流放大电路控制Q_1的基极电流，使得Q_1的基极电流增加，振荡加强，从而使+5V的输出电压得以增加，使+5V电压得到稳定，其他3个电压也将得到调整而取得稳压效果。

在+12V输出端接有可控硅SCR。如因某种原因造成12V电压增高，增高到一定程度，可控硅SCR电路动作，可控硅SCR导通，+12V电压被短路，此时Q_1因负载过重而停止振荡，整个电源停止工作。+12V电压消失后，可控硅SCR又重新截止，电源又重新工作，如此周而复始直至故障被排除。所以当用户听到电源出现滴答滴答的开关声时应立即关闭电源并作认真检查，看看输入交流电压是否过高，或者内部电路是否有断路或短路而造成Q_1工作电流过大等故障。

现举一例。主述误将电源变换开关拨至110V档，接入220V电压后将电源烧毁。经开机检查，发现保险丝(2A)烧断，更换一只后，电源仍不能工作。根据线路分析，苹果机开关电源中110V／220V电源变换电路非常简单，如图仅用一只开关K_2，220V时开关断开，桥式整流器正常工作。交流电源为110V时开关K_2接通，桥式整流器与后面的两只电解电容C_1C_2组成倍压整流，以保证在供电电压为110V时仍能输出同样的直流电压。现误将转换开关接至110V而输入电压仍为220V，就是把220V交流电压

进行倍压整流，直流电压过高，超过了高反压管Q_1所能承担的程度，则Q_1有可能被击穿。经实测桥式整流管输出端有电压300V，说明整流部份无故障。而Q_1集电极上测不到电压，查知R_4(1Ω)电阻被烧毁，这进一步说明Q_1已被击穿，拆下Q_1后测知Q_1果然损坏。Q_1的型号是2SC3153，这是一种塑封的大功率管，一般要配到同型号的管子较困难，可用其他开关电源管或电视机中的行管，只要$V_{CEO}>600V$，$P_{max}>30W$都可代用，实际使用效果良好，可代用的三极管有2SC3505，2SC3150，2SC1358，2SC1325等塑封管。亦可用金属封装的BU208，2SC850等型号，只是调换时要重新钻孔。在有的开关电源中就是使用金属封装的管子BU208，在这种情况下只要直接换上一只好的BU208即可。更换过Q_1后要顺便再检查一下Q_1回路中其他的元件。经测量发现R_7亦被烧毁。把R_4，R_7(1Ω1/2W)电阻更换后(可用电阻丝改制)，接上220V电压后，脉冲电源仍不工作，测量Q_1基极电压为零。检查与基极电压有关的元件，知D_1击穿短路，故偏置电压加不上。另Q_3，Q_4亦被击穿。更换时D_1的要求不高，只要耐压100V以上普通硅二极管即可，Q_3是NPN硅管可用3DG6，2SC9014等管代替，Q_4是PNP硅管可用3CG4，2SC9015等管代替。更换过D_1，Q_3，Q_4后，电源正常工作，说明Q_1被击穿后，直流高压加至D_1，Q_3，Q_4致使D_1，Q_3，Q_4被击穿。电源其他部份，如变压器次级部分因与直流高压隔离，可不作进一步检查。至此苹果机电源检修完毕。



IBM-PC/XT常见故障分析与维修（一）

华南师范大学 张毅忠 郑存陆 罗家耀

本文是以IBM-PC/XT机型为主叙述IBM个人计算机及其兼容机常见故障现象、故障点以及从原理上分析故障。目的是为读者提供维修该类机型所必备的电路知识和维修方法与经验。

IBM机有个开机自检系统（POST），在开机上电时自动执行各部分性能测试；另外，IBM机还提供高级诊断程序（V2.02或V2.05版本），在故障时可以运行此诊断程序，对机器各部分进行测试。有关检测程序使用方法，请参看本刊87年2期。这里仅叙述开机自检测程序的测试顺序。了解自检测部件的顺序对判断故障也是有益的。加电开机后，便进入自检系统，并按下顺序对各个部分进行检测：

▲ 8088处理器测试，检查各标志位、寄存器和条件转移。如果有错，机器暂停。

▲ ROS校验，这对包含BIOS的8KB ROM芯片校验和测试，校验和不为0就出错，机器暂停。

▲ 8237DMA通道寄存器初始化测试，测试出错，机器暂停。

▲ 基本16KBRAM测试，对前32KB RAM写入AA，55，FF，01和00，并读出比较写入与读出是否相符，不相符就出错，系统停止工作。

▲ 6845CRT控制器测试，初始化并启动CRT控制器，检查视屏RAM，若出错喇叭声响系统停运。

▲ 8259A中断控制器测试，读/写中断屏蔽寄存器（全1或全0），检查热中断（不可预料中断）是否发生？出错时显示101错误代码，系统停止工作。

▲ 8253定时器检查，检查8253计时是否会太快或太慢？是，显示101错误代码，系统停止工作。

▲ 键盘测试，总清键盘并检查扫描码AAH有无返回到主机；连续按键检测，出错显示错误代码301。

▲ 扩充I/O盒测试，有I/O扩充盒就测试其相连接的数据和地址总线，若出错，显示1801错误代码。

▲ 附加的存贮器读/写测试，冷启动时才执行，若出错，印出错误信息。

▲ 检查包含有BASIC解释程序的4块ROS，若出错，显示错误信息，并发响声。

▲ 磁盘机连接测试，检测软磁盘控制器（NEC FCD）工作是否正常，启动驱动器并进行检测，若出错，显示错误代码601。如果硬磁盘出错，显示1701。

了解开机自检顺序及其检测内容可以便于我们寻找故障原因，比如，当你发现屏幕显示301，不仅可以判定是键盘方面故障，而且可以断定键盘检查以前所检查的部分都是正常的。

一、开机后系统不启动，无喇叭响声也无显示

这类故障现象的原因很多，也无法使用高级诊断程序，究竟是什么原因？需要作进一步观察、测试和分析。

1.电源故障

是否因电源问题引起，较容易断别，因为机内小风扇是由稳压电源24V（+12V～-12V）来驱动的，所以开机后首先细听机内的小风扇有无转动响声，若有，说明市电已接通，电源大多是正常，若无，就先得检查电源插头是否插牢，保险丝是否烧毁，如果正常就需拆下机盖用万用表测试电源插座（图1）的各脚电压符合图1所示，就可以排除故障来自电源，便可以怀疑故障是由主板上的有关IC芯片损坏所致。

若开机后马上烧毁保险丝，应检查查明短路产生的原因；如果发现电源送到主板的电压不正常，应拆开电源，先观察有无明显的烧坏的元器件，否则按电源的原理图进行检查，直至查出故障原因为止。

PC/XT电源是一种脉冲宽度可调变换器型

稳压电路，其原理图示于图2 。市电先经由电容C1～C4和电感L1～L3组成电网滤波器，然后进入由C5～C8和桥堆DL组成的高压整流器，电网滤波以隔离电网与稳压器间的相互干扰；高压整流滤波是将220伏的交流电变为约300V高压直流电。R1为限流电阻，以减少启动时冲击电流保护高压整流电路。

电容C5～C8作为分压电容，与开关功率管BG1和BG2以高频变压器B组成半桥式的高频变换器，把300V的高压直流变换成宽度可变，频率为20KHz的矩形波。BG1导通关闭与BG2关闭导通交替进行，BG1导通时间决定正脉宽，BG2导通时间决定负脉宽。D1和D2二极管分别并联在BG1和BG2的c e两脚，起嵌位防止功率管击穿。

高频变压器次级，集成块SD1，电感L4～L7，电容C14～C19组成高频整流滤波器，

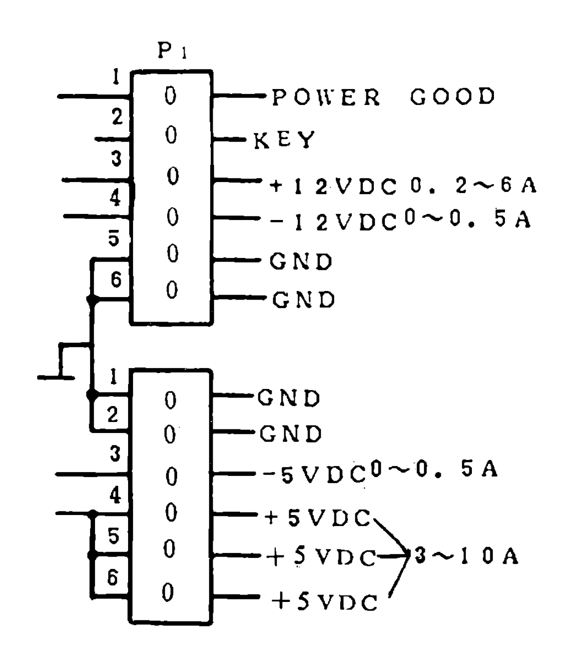


图1 电源插座

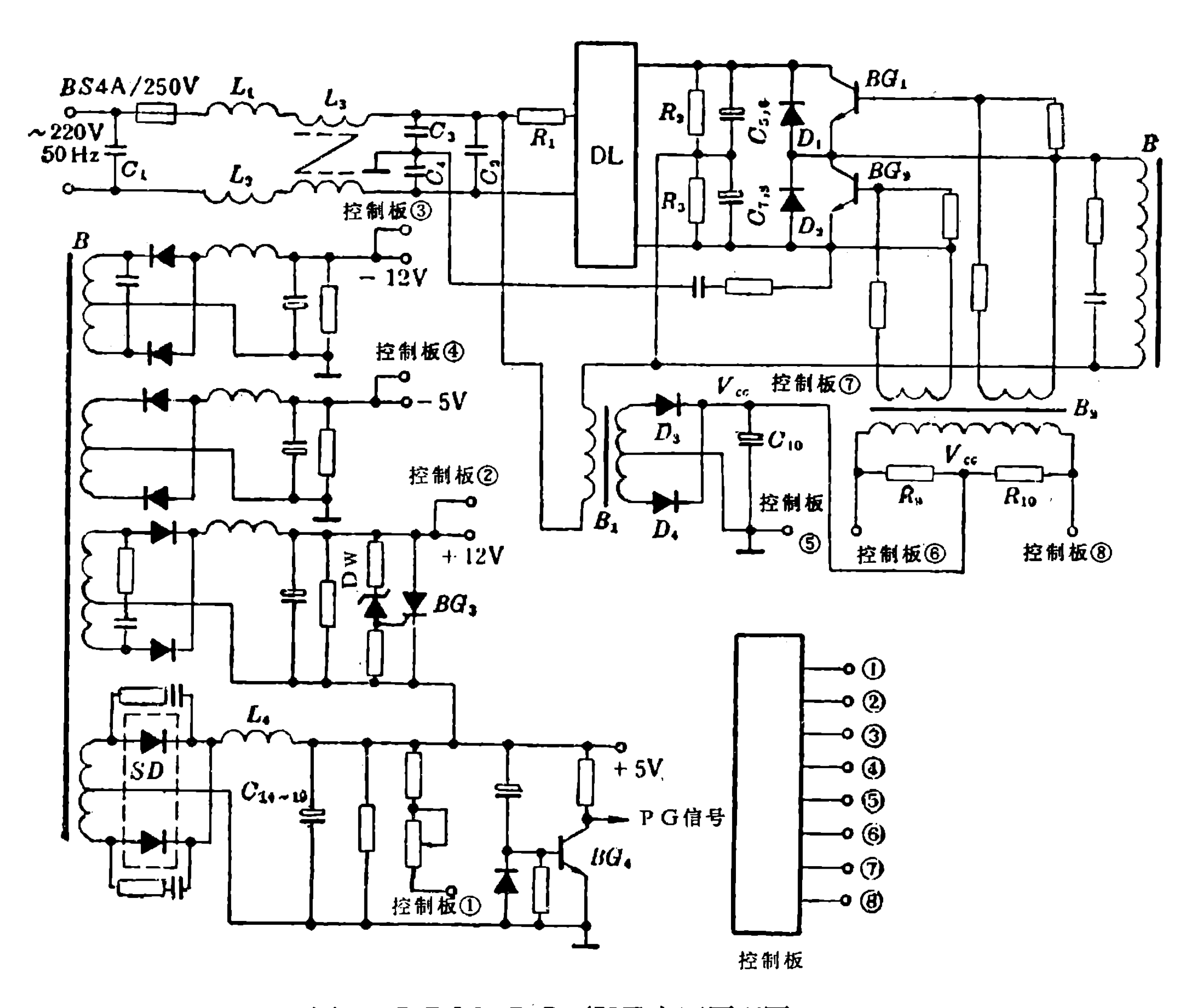


图2 IBM—PC／XT电源原理图

使高频矩形脉冲变为低压直流（+5V，−5V，+12V，−12V）输出。输出电压正常时，同时输出电源好（PG）信号。四级直流电压取样送至控制板。控制板包括：比较放大器，锯齿波发生器，脉宽调制器，分频器，内部稳压器和控制保护电路等。当+5V电压取样①发生变化，经控制板电路作用产生变宽或变窄的脉冲驱动信号由⑥、⑧脚送出，通过驱动变压器B2传到BG1和BG2的基极，改变BG1和BG2的开关时间，从而改变高频脉冲的正、负波宽度，由此改变输出电压平均值的大小，以稳定+5V值。控制板的电源由变压器B1和D3、D4、C10组成的整流滤波电路提供。

可控制硅BG3和稳压管DW组成过电压保护电路，对+12V+5V进行过压保护，而过流保护，各档直流电压的欠压保护都由控制板承担，一旦其中有一档输出电压下跌，整个电源停止工作。

如果，排除是电源故障，就需从下面几方面去寻找故障点。

2·无时钟信号，8284A损坏

IBM微机中央处理器8088及其整个系统都需要正确的时钟信号去驱动，否则系统无法运转。系统中所用到的各种时钟信号都由时钟发生器8284A来产生。图5是系统接线板（1）电路图，在机器合上电源，POWER GOOD（电源好）信号为高电平时，时钟发生器U1（8284A）开始正常工作，它将对石英晶振产生的14·31818MHz时钟信号（脚12，OSC）进行分频：3分频后产生4·733MHz时钟从脚8（CLK88）输出；6分频后产生2·387MHz时钟从脚2（PCLK）输出。用示波器测试8284A的脚12，8，2，均有相应频率的方波。若无波形输出，必然是8284A芯片损坏。（见图3）

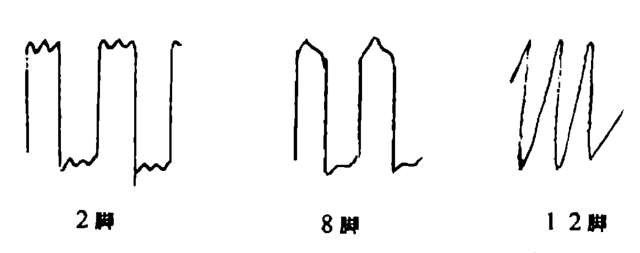


图3 8284A正常工作波形

3·数据总线上波形不正常

如果系统时钟正常，又排除是8088CPU故障的话（CPU故障较少见），就得查试数据总线上的损坏元器件。在开机时复位信号使8088CPU复位，复位后CPU便从$FFFF0单元开始执行指令。从图5可见U2（74LS245）是数据接送器，U5（74LS373）是地址锁存器。CPU的脚9～16是分时传送数据和地址信息（AD0～AD7）用的。由系统地址锁存信号（ALE）控制，将地址信息锁存在U5；数据允许信号（DEN）反相后去控制U2接收和发送数据信息。传送的方向由信号DT/R来控制，即当DT/$\overline{R}$=1时，数据从CPU向外发送；当DT/$\overline{R}$=0时CPU接收外来的数据。可见，U2是个双向数据接送器。

正常情况，开机后数据总线上的数据信息会不断变化，用示波器测试会显现出具有一定幅值，变化无常，密集而断续形象的波形。但是，如果U2，U22损坏将出现以下现象：

①U2损坏时，用示波器测试主板上多数波形都会正常，但数据总线会无波形，或波幅很小呈锯齿形状。U2脚1，19的正常波形见图4所示。

②U22损坏时，其脚12无低位电号输出，使U2不能工作，此时数据总线上D0～D7八根数据线均无波形。

应该指出，U9，U15损坏（图末标，请注意查看以后几期的电路图）也都可能影响到D0～D7数据总线的正常工作，也就是说，当数据总线上波形不正常时，焊脱U2，U22测试均正常的话，就应怀疑U9和U15芯片损坏了。

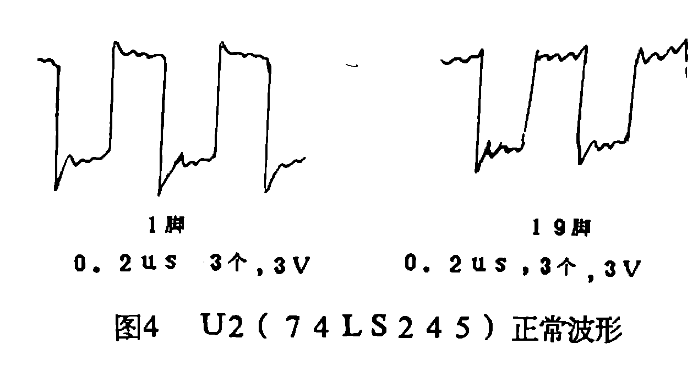


图4 U2（74LS245）正常波形

图 5　系统板图 1

APPLE—Ⅱ 通用数据处理程序

河北师大物理系 冯希章

在Apple机上利用A/D+D/A卡进行数据采集非常方便。但当采集数据较多时则需占较大的内存区，为了节省有效内存区，我们把语言板上的RAM区作为“虚拟”内存来使用。在为了不同目的进行数据采集时（我们是在物理实验中进行），往往希望先大致观察一下采集到的结果是否符合实验要求，若直接从采到的数据值来判断则既费时又费力。我们是将数据值直接在显示器屏幕上显示出相应曲线，由曲线形状来判断数据是否可取，这样很直观又方便。如果数据出入较大则本组数据可舍去，重新采集另一组。如果本组数据可取，则利用打印机将曲线打印出来，即可作为实验报告的一部分。我们在教学过程中，为了实现各种过程的数据采集和各种方法的计算处理编制了不同的程序，但对前面谈到的数据处理（不会计算处理）却可用一个通用程序来完成。

这个通用数据处理系统，除了Apple机的基本系统外，只需再加一块8位A/D+D/A卡即可。

一．本数据处理系统的特点

1．主程序用浮点BASIC语言编写，便于人机对话，也便于一般用户增删和修改程序。

2．子程序用6502汇编语言编写，这是为了加快处理速度。当然，还有下面4种即将谈到的原因。考虑到一般用户不太熟悉这种语言，为使这部分内容尽量简单，本程序采用了在Apple机监控程序中已有的子程序，用CALL指令调用，这样一来使汇编语言程序压缩到最简，很易读懂。

3．为提高微机利用率，数据采集分组进行，每组采集256个数据。每采完一组，首先利用显示屏幕将相应数据的曲线绘出，供用户参改。若认为本组数据可取，则启动打印机将该曲线打印出来，然后将这组数据转储到16K语言卡中，而不转入磁盘中。该数据区留给再次采集数据使用。

4．为有效利用内存，并加快数据采集和处理速度，将语言板上的RAM作“虚拟”内存使用。

虽然一般Apple机上都插有16K语言卡，但在DOS操作系统，运行浮点BASIC语言的情况下，并没有用到这部分RAM区，甚是可惜。考虑到此，为了节约内存，以便存放计算程序及其中间过程生成的数据等。本程序存放采集到的数据（8位数据）只占$FF个固定内存区。经显示和打印及计算处理后，即将每组数据分次转储于$D000以上的RAM区，这部分RAM区起到了“虚拟”内存的作用。这样既扩大了内存使用区又加快了整个数据采集和处理的速度。这是因为，虽然采集数据的存放区只占$FF个内存，但多次采集时，前一组采到的数据不是转存入磁盘中而是转储到语言板上的RAM中，省去和磁盘频繁打交道，故速度大大加快。$FF个内存区起了转运站的作用，同时又利用语言板上的RAM区作全部数据的暂存区。当数据全部采集和处理完毕，统一将存于$D000以上内存区的数据转入主板内存区，然后再转储入磁盘中，作为数据文件保存。

由于$D000的ROM区中固化了浮点BASIC语言的解释系统，所以用户不能直接用浮点BASIC语言对语言板上的RAM区进行存取操作，只好选用6502汇编语言进行。当然，这就要求注意软开关的设置。

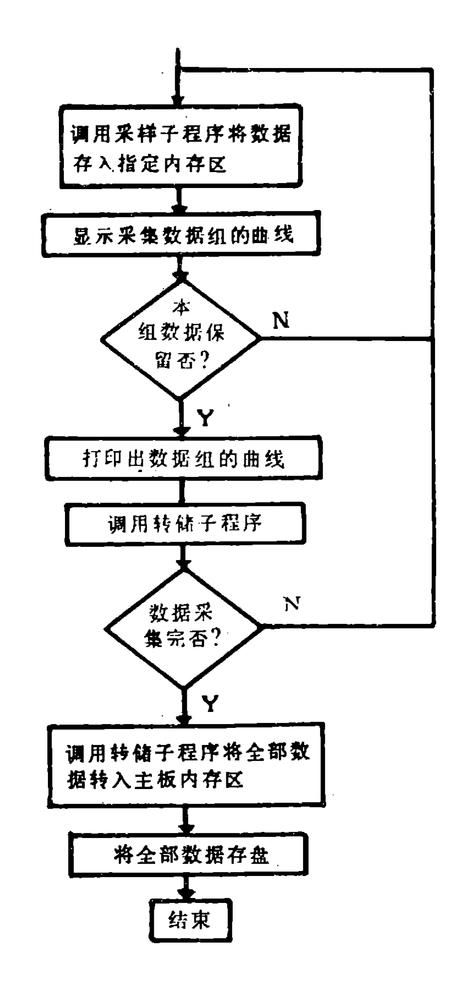
5．显示所采集的数据组曲线，我们采用的是高分辨度图形第一页和文本混合显示方式，留下四行文字窗口用于进行人机对话。

二．软件结构

1．主程序框图：

2．程序介绍：

下面所附的程序清单就是按照上面谈到的特点而编制的通用数据处理程序。为了叙述简单，对于程序中必要的注释及换块之间的连接等都作了相应



简化，以节省篇幅，这些请读者注意。

数据采集子程序由于所采用的A／D转换板及用处不同，编制的程序也不一样，本文不涉及。本程序的第20行是为了调试程序而给出的模拟数据采集，将采集到的数据存入内存$8000(32768)～$80FF中。在实际进行数据采集时，将数据采集子程代替本程序的这部份程序即可。

当采集每个数据的时间大于30秒，建议采用中断方式采集。当采集频率较高时，在处理数据转储过程中会丢掉一些数据。若这些数据不允许丢掉，则每组采集数据个数可按需要作相应调整。用汇编语言进行数据采集速度很快，每组多采几个数据不会影响整个数据处理的速度。

从100行开始就是有关数据处理程序。

程序10行中的C是计采集数据组的次数。

程序100～140行是将数据的曲线显示于屏幕上。

程序150～170行进行人机对话。

程序200～220行是将数据曲线由打印机打印输出。

程序300～360行为转储子程序，是将数据从主板内存区$8000转储到语言板内存区$D000以上区域。用汇编语言写成。

```
0300-  AD 81 C0   LDA $C081
0303-  AD 81 C0   LDA $C081
0306-  A0 00      LDY #$00
0308-  4C 2C FE   JMP $FE2C
```

其中$FE2C是监控程序中MOVE子程序的入口。

程序370～390行进行人机对话并控制每采够1024个数据后程序即自动转入存盘程序。

程序400～450行也是转储子程序，是将全部采集到的数据从$D000～$D000+256*C 转储到以$5000为起始地址的主板内存区。用汇编语言写成。

```
030B-  AD 83 C0   LDA $C083
030E-  A0 00      LDY #$00
0310-  20 2C FE   JSR $FE2C
0313-  AD 81 C0   LDA $C081
0316-  60         RTS
```

在以上两个转储子程序中都注意到软开关的设置。在前一个子程序中二条LDA $C081是使可以写语言板上的RAM区，而后一个子程序中的一条LDA $C081是转为读ROM区的软开关，转换后才返回主程序，以便运行浮点BASIC程序。

程序460行为存盘命令，以二进制文件的形式将全部采集到的数据存入磁盘。

本程序规定每采够1024个数据后即自动转入存盘程序，每执行完一次后，若还要再采集另一组数据，则要注意更改460语句行中的文件名称。若拟改变转入存盘时的数据个数(256的整倍数)，需改390和460程序行的有关内容。当采集数据的个数每次要求都不一样时，则需更改460程序行，将数据以文本文件方式存盘。

整个程序均在Apple－II机上运行通过。

```
10 S = 32768:C = 0
20  FOR I = 0 TO 255: POKE S + I,
    I: NEXT
100 D$ =  CHR$ (4): HGR : HCOLOR=
    3
110  HPLOT 0,120 TO 279,120
120  FOR I = 0 TO 255
130  HPLOT I,120 -  PEEK (S + I)
     3 TO I + 1,120 -  PEEK (S +
     I + 1) / 3
140  NEXT
150  HOME : VTAB 21
160  INPUT "DATA AVAILABILITY?(Y/
     N)--";Y$
```

PASCAL语言在结构化程序设计中的应用

上海纺织工业设计院　桂业冲

PASCAL语言诞生十多年来，已广泛应用于程序设计的教学，PASCAL语言与BASIC语言同样易于学习与使用，但功能比BASIC语言强得多。本文通过介绍应用PASCAL语言编写结构化程序的方法，说明了近年来流行的结构化程序设计的优越性。文中对结构化程序设计作了论述，列举了实例来详细阐述PASCAL语言进行结构化程序设计的具体步骤及过程。

一．引　言

PASCAL语言是七十年代初在ALGOL 60语言的基础上发展起来的一种结构程序设计语言，它具有语句结构化和数据结构化的特点。PASCAL语言功能强，数据类型丰富，写出的程序简洁易读，适用于数值和非数值问题的描述，既能用于应用程序设计，亦能用于系统程序设计。PASCAL语言象BASIC语言一样易于学习和使用，但在支持结构化程序设计和表达算法的能力上比BASIC语言优越得多，它与APL语言同样有效地表达算法，但具有更好的易读性。因此，PASCAL语言已成为目前最流行的几种程序设计语言之一。在欧美国家中，越来越多的大学选用PASCAL语言作为教学的第一计算机语言。

二．结构化程序设计的内容

对于一个大型的程序系统，怎样判断所写程序的正确性，常常是十分困难的。结构化程序设计就是为了使程序具有合理的结构，以保证程序尽可能正确，并使其正确性验证比较容易，因而规定了一套程序设计的准则，按此准则设计出来的程序称为“结构化程序”。PASCAL语言是第一个支持这种概念而设计出来的结构化程序设计语言，其语言成份中即反映了结构化程序设计的要求和限制所以能方便地用来书写结构化程序。因此，用PASCAL语言写出的大型程序易于保证其程序的正确性，亦便于检验及修正错误。

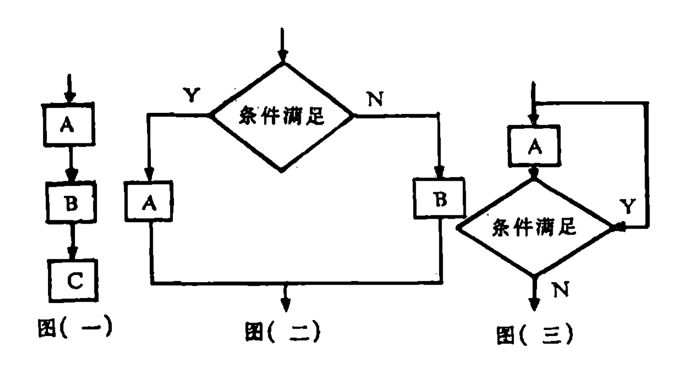
结构化程序设计是一种采用结构化的方式编制程序的方法。这种方法不是一步就编制成可执行的程序，而是分步进行，逐步求精。第一步编写出的程序的抽象级最高，第二步编写出的程序的抽象级比第一步低，依此类推，第I步编出的程序的抽象级比第I－1步低，直到最后，第n步编写出的程序即为可执行的程序。采用这种方法编制程序，看起来似乎复杂，其实好处很多，可使程序易读、易写、易调试、易维护、易于保证程序的正确性，易

```
170  IF Y$ = "N" THEN 20
200  PR# 1: POKE 1913,1
210  PRINT  CHR$ (17)
220  PR# 0
300  RESTORE
310  FOR I = 0 TO 10: READ A: POKE
     768 + I,A: NEXT
320  DATA    173,129,192,173,129
     192,160,0,76,44,254
330  POKE 60,0: POKE 61,128
340  POKE 62,0: POKE 63,129
350  POKE 66,0: POKE 67,208 + C
360  CALL 768
370  INPUT "CONTINVE DATA ACQUISI
     TION?(Y/N)--";Y$
380  IF Y$ = "N" THEN 400
390 C = C + 1: IF C < 4 THEN 20
400  FOR K = 0 TO 11: READ A: POKE
     779 + K,A: NEXT
410  DATA  173,131,192,160,0,32,4
     4,254,173,129,192,96
420  POKE 60,0: POKE 61,208
430  POKE 62,0: POKE 63,209 + C
440  POKE 66,0: POKE 67,80
450  CALL 779
460  PRINT D$;"BSAVE FANG,A$5000,
     L1024"
470  END
```

于验证程序的正确性。特别是，在软件工程中采用这种方法，可以收到较好的效果。所以将这种方法称之为“自顶向下”或“逐步求精”法。

具体地说，结构化程序设计的方法是，首先提出解决问题的办法，然后再逐步改进并完善这个解决办法，直到获得一个可执行的程序为止。因此算法设计的第一步只是用文字或公式描述，没有细节，也不要考虑对象如何表示；然后将其进一步分解成若干个子任务。继而类似地对这些子任务再进行分解，直到能用PASCAL语言来描述这些子任务为止。

结构化程序设计方法规定以下三种基本结构：1·顺序结构；2·分支结构；3·循环结构。分别如图1、图2、图3所示。



图(一)　图(二)　图(三)

三．结构化程序设计的过程

我们通过举例来说明PASCAL语言进行结构化程序设计的过程，下面的实例是求两多项式a与b之和，设其和为多项式c。

第一步：

```
PROCEDURE          polyadd;
BEGIN
   c ← a + b
END;
```

第二步：若多项式有下述形式

$a_n x^n + a_{n-1} x^{n-1} + \cdots + a_1 x + a_0$ 其中 $a_n \neq 0$，可以将n，a_n，a_{n-1}，… a_1，a_0 构成一个一维数组，用它来表示该多项式。显然第一个元素是多项式的次数，以后的元素是按递降方幂次序排列的系数。若多项式a的次数为n，多项式b的次数为m(设n >m)，则其和多项式c为n次。有了这些约定后，则数组a、b的元素分别为：n，a_n，……，a_0；m，bm，……，b_0，n与m均为一确定的常数。

```
PROCEDURE          polyadd;
VAR                a,c : ARRAY [1..n] OF integer;
                   b : ARRAY [1..m] OF integer;
BEGIN

将a 中大于m之系数传送给C；
将a 与b 中幂相同的系数相加结果传送C

END;
```

第三步：将上述两个子任务再分别用PASCAL语言写出，即可得到下面可执行的子程序：

```
PROCEDURE          polyadd;
VAR                a,c : ARRAY [1..n] OF integer;
                   b : ARRAY [1..m] OF integer;
                   i,j : integer;
BEGIN
  i:=1;
  REPEAT
    c[i]:=a[i];
    i:=i+1
  UNTIL n-i+1 < m;
  j:=2;
  WHILE i <> n+1 DO
    BEGIN
      c[i]:=a[i]+b[j];
      i:=i+1;
      j:=j+1
    END
END;
```

说明：本程序在DUAL／68000，IBM PC／XT，IBM5550，APPLEⅡ，TRS－80等微型机中均可运行。

当然，上述多项式的表示方法对于形如 $X^{1000}+1$ 的多项式是不恰当的，因为它需要用具有1002个元素的一维数组来表示，这显然要耗费众多的存贮空间，对此还需要用另外的多项式表示方法，即用一维数组来保存它的非零系数。

四．结束语

“逐步求精”的程序设计方法，可以说是结构化程序设计的核心。从上例中可以看出，这种方法要求从问题的全局出发，然后一步一步地求精，每一步求精的正确性都以前一步的正确性为前提，这样一步一步地继续求精下去，直到构造出以PASCAL语言表示的程序为止。因为每一步求精都以前一步的正确性为前提，所以这种方法容易构造出正确的程序。当然，这种方法由于强调先整体后局部，故要求程序设计者对问题的全局结构必须有较透彻的理解，而这对初学程序设计的人往往是困难的，但通过不断地实践，亦是能够正确地掌握并熟练地运用的。

万有引力作用下的抛体运动

汕头金山中学　　何管略

在地面附近，以不大的初速度作抛体运动，可将重力看作大小、方向都不随时间改变的恒力，情况比较简单。在高空，以较大的初速度作抛体运动（不计空气阻力），物体所受万有引力的大小和方向，都随它所在的位置而改变，运动情况复杂得多。运用苹果机的HGR功能，分析万有引力作用下抛体运动的规律，有助于我们加深理解人造地球卫星和各种航天器的飞行情况。

附件①是屏幕显示的地球模型。在高度为H处，有物体M，以初速度V和抛射角A（从水平向右的轴线开始，反时针方向为正）抛出。

附件②列表提供九种不同的初速度和抛射角：其中有竖直上抛、自由落体、平抛（3，4，5，6，9）和斜向上抛（7，8）。

附件③是屏幕显示上述九种运动的相应轨迹；其中有直线，抛物线（9），圆（5）和椭圆（4，6，7是完整的；3，8是不完整的）。

附件④是用随机函数取定一个高空位置之后，通过键盘输入适当大小和方向的初速度，使运动轨迹既不接触地球表面，又不超越屏幕范围，从而得到一个完整的椭圆。用电子游戏的形式来显示科学内容，将规律性的知识融化在文娱活动之中，颇能引人兴趣。

附件⑤是用浮点BASIC语言编写的程序清单。程序设计的主要思路是：已知物体的位置坐标（x，y）和速度分量（V_x，V_y），在万有引力（不计空气阻力）作用下，可求出经短时间之后的加速度分量，新的速度分量（V_x'，V_y'）以及新的位置坐标（x'，y'）。如此反复计算、作图，就可描绘出物体的运动轨迹。具体证明如下：

设地球质量为M，位于坐标原点。今有质量为m的物体，位于坐标（x，y）处，运动速度分量为（V_x，V_y）。显然，M与m之间的距离$r=\sqrt{X^2+y^2}$，m所受的万有引力$F=-GMm/r^2$，该负号表示力的方向从m指向坐标原点。取r与X轴之间的夹角为θ（从X轴开始，反时针向为正），则此力在X轴方向的分力$F_x=F\cos\theta=-GMm/r^2\cdot X/Y$，物体的加速度$a_x=F_x/m=-kx/r^3$。在第一个短时间t之内，物体的平均速度$v_x'=v_x+a_xt/2$，在第一个t之后，物体的新位置坐标$x'=x+v_x't$。

同理，我们可写出$a_y=-ky/r^3$，$v_y'=v_y+a_yt/2$，$y'=y+v_y't$。

至于第二个t以后新位置坐标，算法理应有所不同：距离$r'=\sqrt{x'^2+y'^2}$，加速度$a_x'=-kx'/r'^3$，速度$v_x''=v_x'+a_x't$（注意，从此以后，不必再用t/2了），位置坐标$x''=x'+v_x''t$。

实践是检验理论的标准。屏幕显示的圆形和椭圆形轨迹，能够准确封闭，而且具有两条互相垂直的对称轴线，就是上述算法误差甚小的证明（333——566句）。

为了提高屏幕显示效果，我们采取下列措施：

⑴用小圆点（子程序1111——1155句）表示物体的位置。轨迹触地或到达屏幕边界时，原处的小圆点消失，在轨迹终端显示新的小圆点，同时发出"B"声（777句）。

⑵作图过程中，按任意键，可暂停运行，小圆点从原处移到轨迹终端（433句），屏幕下方显示菜单。如按C键，恢复全屏幕图象，继续运行子程序。

⑶物体作竖直上抛或自由落体时，运动轨迹互相重迭。运动轨迹在小圆点的范围之内时，屏幕不应显示轨迹图象，以保持小圆点的清晰。为了表示物体运行轨迹的连续性，在上述范围内，不断发出"D"声（444句）。

⑷运动轨迹从HGR区到达下方边界时，自动将四行文本区改为全屏幕图象显示（566句）。

（附件⑤）

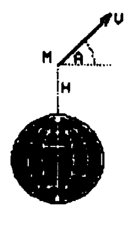
```
11   HGR : HCOLOR= 3:X = 279:Y = 191: HPLOT X,Y: HPLOT X,0: HPLOT
      0,Y: HPLOT 0,0:R = 27:Z = 95: GOSUB 1222
33   HCOLOR= 0: FOR K = 10 TO 19 STEP 9:R = (27 * 27 + K * K) /
      K / 2:R =  INT (R + .5)
44   FOR A = 0 TO 1.2 STEP 1 / R: GOSUB 1444: HPLOT Z - R + K +
```

```
     M,Z - N: HPLOT Z - R + K + M,Z + N: HPLOT Z + R - K - M,
     Z + N: HPLOT Z + R - K - M,Z - N: NEXT A,K
55   FOR R = 33 TO 66 STEP 33:K = 9: IF R = 33 THEN K = 18
66   FOR A = 0 TO .6 STEP 1 / R: GOSUB 1444: HPLOT Z + N,Z - K
      - R + M: HPLOT Z - N,Z - K - R + M: HPLOT Z - N,Z + K +
     R - M: HPLOT Z + N,Z + K + R - M: NEXT A,R
77 R = 27: HPLOT Z,Z - R TO Z,Z + R: HPLOT Z - R,Z TO Z + R,Z

99   HCOLOR= 3: HPLOT Z,Z:R = 29: GOSUB 1222: GOSUB 1322
111   DATA 4.6,90,,,3,,4.6,,5.6,,6.1,,6.1,30,6.1,45,8,
222   HOME : VTAB 24: PRINT ,"R: READ DATA",,"I: INPUT V, A",,
     "P: PRINT HGR ";: GET R$: ON R$ <  > "R" AND R$ <  > "P"
      AND R$ <  > "I" GOTO 999: IF R$ = "P" THEN  GOSUB 2333:
      GOTO 222
244   HCOLOR= 0: ON R$ = "R" GOSUB 1333: ON R$ = "I" GOSUB 132
     2:K =  - 1946:T = .2
255   FOR J = 1 TO 9: HOME : VTAB 24: POKE  - 16301,0:X = 0:Y =
     62: IF J > 1 AND R$ = "I" THEN X = 60 + 120 *  RND (1):Y
      = 120 *  RND (1) - 60
266   GOSUB 3333:C = 3: GOSUB 1111: IF R$ = "R" THEN  READ V,A
     : PRINT J")","H= 6.4","KM",,"V= "V,"KM/S",,"A= "A,"DEG "
     ;: GOTO 311
288   PRINT "   IF V< 0 THEN NEXT": PRINT : INPUT "INPUT V, A:
      ";V$,A$:V =  VAL (V$):A =  VAL (A$): ON V < 0 GOTO 888
311 A = A *  ATN (1) / 45:VX = V *  COS (A):VY = V *  SIN (A)
     :H = 0:HH = 0
333 R =  SQR (X * X + Y * Y): IF R < 31 THEN X = XX:Y = YY: GOTO
     777
433   IF  PEEK ( - 16384) > 127 THEN  POKE  - 16368,0:C = 0: GOSUB
     1111:C = 3: GOSUB 1111: GOSUB 2222
444 R = R * R * R: IF VX = 0 OR ( ABS (M - MM) < 3 AND  ABS (
     N - NN) < 3) THEN D =  PEEK ( - 16336)
455 TT = T: IF HH = 0 THEN HH = 1:TT = T / 2
555 A = K * X / R:VX = VX + A * TT:XX = X:X = X + VX * T: IF
      ABS (M - MM) > 2 OR  ABS (N - NN) > 2 THEN  HPLOT M,N
566 A = K * Y / R:VY = VY + A * TT:YY = Y:Y = Y + VY * T: GOSUB
     3333: IF N > 157 AND H = 0 THEN H = 1: POKE  - 16302,0
666   ON M > 2 AND M < 277 AND N > 2 AND N < 189 GOTO 333
777 C = 0: GOSUB 1111:C = 3: GOSUB 1111: CALL  - 198: GET T$:
      GOSUB 2222
888   IF J < 9 THEN C = 0: GOSUB 1111
988   NEXT J
999   HOME : VTAB 11: PRINT ,"GOOD-BYE!": TEXT : END
1111   HCOLOR= C: IF C = 0 THEN M = MM:N = NN
1122   FOR Z = M - 1 TO M + 1: HPLOT Z,N - 2 TO Z,N + 2: NEXT
     : HPLOT M - 2,N - 1 TO M - 2,N + 1: HPLOT M + 2,N - 1 TO
     M + 2,N + 1
1133   IF C = 3 THEN  HCOLOR= 0: HPLOT M,N - 1 TO M,N + 1: HPLOT
     M - 1,N TO M + 1,N:MM = M:NN = N
1144   HCOLOR= 3: IF C = 3 OR V >  = 0 THEN  HPLOT M,N
1155   ON C = 0 GOSUB 3333: RETURN
1222   FOR A = 0 TO 1.6 STEP 1 / R: GOSUB 1444: HPLOT Z + M,Z -
     N: GOSUB 1288: HPLOT Z + M,Z + N: GOSUB 1288
```

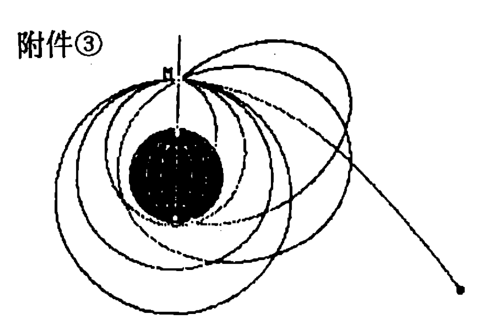
```
1233  HPLOT Z - M,Z + N: GOSUB 1288: HPLOT Z - M,Z - N: GOSUB
      1288: NEXT : RETURN
1288  IF R = 27 THEN  HPLOT  TO Z,Z
1299  RETURN
1322 M = 88:N = 28: HPLOT M - 2,N - 3 TO M - 2,N + 3: HPLOT M
      + 2,N - 3 TO M + 2,N + 3: HPLOT M,N - 1 TO M,N: HPLOT M
      - 1,N - 2: HPLOT M + 1,N - 2
1333 M = 123:N = 5: HPLOT 95,33 TO M,N: HPLOT M + 2,N + 2 TO
      128,0 TO M - 1,N - 1
1344 M = 133:N = 3: HPLOT M - 2,N - 3 TO M - 2,N + 1: HPLOT M
      + 2,N - 3 TO M + 2,N + 1: HPLOT M - 1,N + 2: HPLOT M +
      1,N + 2: HPLOT M,N + 3
1355 M = 108:N = 28: HPLOT M - 2,N - 1 TO M - 2,N + 3: HPLOT
      M + 2,N - 1 TO M + 2,N + 3: HPLOT M - 1,N + 1 TO M + 1,N
      + 1: HPLOT M - 1,N - 2: HPLOT M + 1,N - 2: HPLOT M,N -
      3
1366 M = 100:N = 48: HPLOT M - 2,N - 3 TO M - 2,N + 3: HPLOT
      M + 2,N - 3 TO M + 2,N + 3: HPLOT M - 1,N TO M + 1,N
1377 R = 23: FOR A = 0 TO  ATN (1) STEP 2 / R: GOSUB 1444: HPLOT
      95 + M,33 - N: NEXT : FOR A = 0 TO 30 STEP 2: HPLOT 98 +
      A,33: HPLOT 95,34 + A: NEXT : RETURN
1444 M =  INT (R *  COS (A) + .5):N =  INT (R *  SIN (A) + .5
      ): RETURN
2222  HOME : VTAB 24: POKE  - 16301,0: PRINT ,"C: CONT",,,"N:
      NEXT V, A",,"P: PRINT HGR",,"E: END ";: GET T$: PRINT :
      ON T$ = "E" GOTO 999: IF T$ = "N" THEN  POP : GOTO 888
2233  IF T$ = "C" THEN  POKE  - 16302,0: RETURN
2244  ON T$ = "P" GOSUB 2333: GOTO 2222
2333  PR# 1: PRINT : POKE 1913,1: PRINT  CHR$ (17): PR# 0: RETURN

3333 M = 95 +  INT (X + .5): IF M < 2 THEN M = 2
3344  IF M > 277 THEN M = 277
3355 N = 95 -  INT (Y + .5): IF N < 2 THEN N = 2
3366  IF N > 189 THEN N = 189
3377  RETURN
```

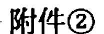


附件①

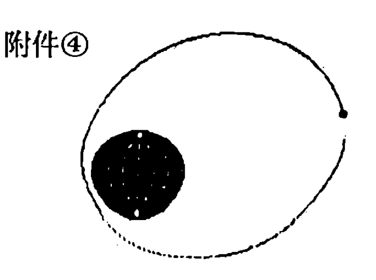
附件③



NO.	H:KM	V:KM/S	A:DEG
1)	6.4	4.6	90
2)	6.4	0	0
3)	6.4	3	0
4)	6.4	4.6	0
5)	6.4	5.6	0
6)	6.4	6.1	0
7)	6.4	6.1	30
8)	6.4	6.1	45
9)	6.4	8	0


附件②

附件④



微电脑摸拟条件反射的建立

湖北省沙市 第三中学 王秋远 杨瑞华

讲解条件反射的建立，关键是非条件刺激引起大脑皮层一定区域形成的兴奋灶和条件刺激引起大脑皮层一定区域形成的兴奋灶之间在功能上的暂时接通而发生的暂时联系。用微电脑摸拟条件反射的建立，则是通过鲜明的图象、可见的“神经冲动”、“兴奋传递”、“兴奋灶的联系”形象地进行演示、效果良好、现简介如下：

一、程序说明

5 行：程序名称：条件反射的建立

1 0 —8 1 行：绘狗头部形态、结构图及实验装置——唾液漏斗图

1 0 0 —5 0 6 行：摸拟非条件（食物）刺激引起非条件反射（唾液分泌）并在大脑皮层一定区域（味觉中枢）形成兴奋灶

5 1 2 —7 6 0 行：摸拟条件（灯光）刺激引起大脑皮层一定区域（视觉中枢）产生兴奋灶。

5 1 2 —7 6 0 、3 5 0 —5 0 6 、1 8 0 —2 9 5 行：摸拟条件刺激与非条件刺激多次结合而建立条件反射。

二、课堂演示与电脑操作

1 · 先将程序（简称T I AO J I AN）输入A P P L E —Ⅱ型微电脑内、存入磁盘机中。

2 · 按RUN TI AO J I AN；回车，即呈现狗头部形态结构示意图及实验装置——唾液漏斗图。

3 · 结构图呈现后，即有光标闪动、按任意一键，电脑则摸拟非条件反射过程：即食物（非条件）刺激味觉感受器，其冲动沿着传入神经传到唾液分泌中枢、由此发出神经将冲动传到唾液腺，引起唾液分泌，唾液经唾液漏斗流出。同时兴奋由唾液分泌中枢经侧支传异通路上行到大脑皮味觉中枢形成兴奋灶。若要重复这一过程，待屏幕上出现提示后，再按 S HI 回车，即可。

4 · 非条件反射结束时，出现提示：S I S H I HUAN S I C HUAN DEN，再按DE N，则是灯光（条件）刺激狗眼内视觉感受器，其冲动沿着传入神经传送到视觉中枢形成兴奋灶。若要重复，待提示出现，按DE N，即可。

5 · 提示S I S HI HUAN S I C HUAN DE N出现后按C，则形成条件反射的建立训练过程。当条件反射形成的训练成熟后，则出现提问：“S I S HI HUAN S I DEN HO T J I AN· 按下T，回车即显示条件反射的建立，如按S H I，则重复非条件反射过程，按DE N则重复灯光（条件）反射的过程。

以上每一个过程请见摸拟图一、二、三、四。

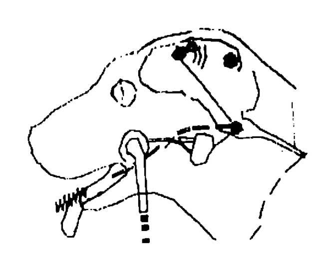


图 2 非条件食物时各兴奋灶传递

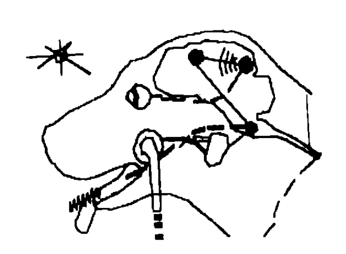


图 3 形成条件时各兴奋灶的连接

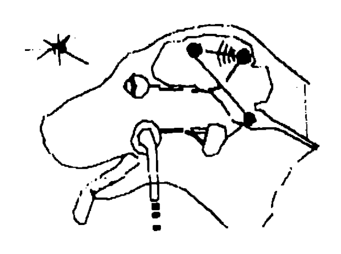


图 4 条件形成时的各兴奋灶的传送

```
5   REN  TIAO JIAN FAN SER DER JIAN LI
10   HGR : HCOLOR= 3
20   HPLOT 255,55 TO 240,43 TO 225,31 TO 210,15 TO 200,7
     TO 190,3 TO 180,2 TO 160,3 TO 140,6 TO 130,9 TO 120,
     14 TO 110,20 TO 96,23 TO 86,31 TO 80,40
25   HPLOT 80,40 TO 70,48 TO 40,60 TO 25,70 TO 18,72 TO
     15,80 TO 18,90 TO 17,92 TO 22,102 TO 25,102 TO 25,
     105 TO 32,112 TO 45,116 TO 55,115 TO 60,114 TO 85,
     106 TO 89,105 TO 95,106 TO 99,109 TO 100,115
28   HPLOT 100,115 TO 80,128 TO 57,138 TO 59,140 TO 60,
     141 TO 70,142 TO 90,137 TO 106,131 TO 123,131 TO
     140,138 TO 160 TO 160,155 TO 175,169
30   HPLOT 205,170 TO 215,160: HPLOT 215,155 TO 220,142:
     HPLOT 225,138 TO 235,125: HPLOT 240,120 TO 260,105
32   FOR Y = 100 TO 55 STEP -2:X = 250
35   HPLOT X,Y: NEXT Y
40   HPLOT 89,105 TO 80, 116 TO 55,129 TO 48,135 TO 40,
     150 TO 41,156 TO 44,160 TO 49,160 TO 51,158 TO 59,
     140
45   HPLOT 57,138 TO 56,134 TO 60,136: HPLOT 46,135 TO
     64,133 TO 66,134 TO 68,132: HPLOT 73,131 TO 73,129
     TO 75,131 TO 75,128 TO 78,129
50   HPLOT 258,105 TO 235,90 TO 220,82 TO 210,74 TO 205,
     73 TO 205,69 TO 216,67 TO 221,57 TO 217,39 TO 209,
     39 TO 205,38 TO 204,37 TO 207,35
55   HPLOT 207,35 TO 204,20 TO 190,12 TO 166,9 TO 160,11
     TO 148,12 TO 145,14 TO 134,14 TO 128,16 TO 116,35
     TO 118,44 TO 130,51 TO 140,52 TO 153,51 TO 158,57
     TO 167,59
58   HPLOT 167,59 TO 176,67 TO 186,83 TO 193,85 TO 200,
     89 TO 210,89 TO 220,89 TO 230,93 TO 258,109
60   HPLOT 157,105 TO 164,84 TO 170,84 TO 178,87 TO 178,
     92 TO 179,95 TO 173,103 TO 170,109 TO 161,111 TO
     157,105
68   HPLOT 158,101 TO 150,92 TO 140,88 TO 120,88: HPLOT
     159,100 TO 150,90 TO 140,86 TO 120,86: HPLOT 162,90
     TO 150,90 TO 148,86 TO 162,88
70   HPLOT 108,106 TO 100,104 TO 96,100 TO 94,95 TO 100,
     81 TO 105,78 TO 114,80 TO 119,85 TO 121,93 TO 116,
     102
75   HPLOT 115,145 TO 109,145 TO 107,100 TO 102,96 TO
     101,98 TO 100,90 TO 105,85 TO 112,86 TO 116,98 TO
     115,145
80   HPLOT 93,54 TO 90,50 TO 92,46 TO 93,54 TO 100,60 TO
     105,60 TO 111,55 TO 113,50 TO 110,44 TO 105,41 TO
     100,40 TO 93,46
81 HPLOT 98,41 TO 101,50 TO 98,59
82   GET A$
100   FOR I = 0 TO 20
104   HCOLOR= 3
105   IF  INT  (I/2) < > I/2 THEN 120
110   HCOLOR= 4
120   HPLOT 36,142 TO 36,134 TO 38,142 TO 40,132 TO 42,
      141 TO 44,129 TO 46,140 TO 48,128 TO 50,138 TO 52,
```

```
      127 TO 54,136 TO 56,124 TO 58,134 TO 60,124 TO 62,
      132 TO 64,124
160   HPLOT 86,115 TO 105,108: HPLOT 86,117 TO 105,110:
      HPLOT 117,103 TO 130,94: HPLOT 117,101 TO 130,92:
      HPLOT 135,94 TO 144,85: HPLOT 135,94 TO 144,85
165   HPLOT 147,83 TO 158,80: HPLOT 147,81 TO 158,78:
      FOR X = 161 TO 175: FOR Y = 80 TO 78 STEP - 1:
      HPLOT X,Y: HPLOT X+20,Y: NEXT Y,X
166   IF A$ = "C" THEN 584
170   NEXT I
180   FOR K = 0 TO 27
185   HCOLOR= 4
186   IF K = 27 THEN 260
190   IF  INT (K / X) < > K/2 THEN 198
195   HCOLOR= 3
198   HPLOT 195,83 TO 196,77 TO 197,85 TO 198,76 TO 199,
      86 TO 200,76 TO 201,86 TO 202,77 TO 203,84 TO 204,
      78 TO 205,83
260   IF "T" <  > B$ THEN HPLOT 195,75 TO 155,28
261   HPLOT 195,85 TO 178,96
290   HPLOT 110,148 TO 116,148 TO 110,149 TO 116,150 TO
      110,151 TO 116,152 TO 110,152: HPLOT 110,155 TO
      116,155 TO 110,156 TO 116,157 TO 110,158 TO 116,
      159 TO 110,159
291   HPLOT 110,165 TO 116,165 TO 110,TO 116,167 TO 110,
      168 TO 116,168
295   NEXT K
300   IF B$ = "T" THEN 770
350   FOR K = 0 TO 27
360   HCOLOR= 4
362   IF K = 27 THEN 500
370   IF  INT (K / 2) <  > K / 2 THEN 390
380   HCOLOR= 3
390   HPLOT 148,23 TO 149,19 TO 150,26 TO 151,17 TO 152,
      28 TO 153,16 TO 154,28 TO 155,16 TO 156,28 TO 157,
      17 TO 158,27 TO 159,18 TO 160,25 TO 161,23 TO 159,
      17
500   HPLOT 163,20 TO 164,23 TO 161,27: HPLOT 167,19 TO
      168,23 TO 165,30: HPLOT 171,17 TO 172,24 TO 169,
      32: HPLOT 176,16 TO 177,25 TO 172,35
501   IF B$ = "T" THEN HPLOT 155,28 TO 195,75
505   NEXT K
506   IF B$ = "T" THEN 180
511   PRINT "SI SHI HUAN SI C HUAN DEN ";
512   INPUT A$
513   IF A$ = "SHI" THEN 100
514   IF A$ = "DEN" THEN 550
515   IF A$ = "C" THEN 100
550   FOR I = 0 TO 25
581   HCOLOR= 4
582   IF  INT (I / 2) <  > I / 2 THEN 584
583   HCOLOR= 3
584   HPLOT 30,16 TO 31,11 TO 35,10 TO 39,11 TO 41,15 TO
      39,19 TO 35,20 TO 31,19 TO 30,16: HPLOT 35,15 TO
```

```
     60 30 TO 22,7: HPLOT 35,15 TO 45,5 TO 15,24
585  HPLOT 35,15 TO 55,15 TO 5,15: HPLOT 35,15 TO 35,3
     TO 35,30
590  HPLOT 98,42 TO 100,50 TO 98,58: HPLOT 97,43 TO 99,
     50 TO 97,57: HPLOT 96,44 TO 98,50 TO 96,56
591  FOR X = 112 TO 122: FOR Y = 50 TO 51: HPLOT X,Y:
     HPLOT X + 10,Y + 3: HPLOT X + 20,Y + 4: NEXT Y,X
592  HPLOT 150,57 TO 160,56: HPLOT 150,56 TO 160,55:
     HPLOT 165,57 TO 175,54: HPLOT 165,56 TO 175,53:
     HPLOT 181,50 TO 193,35: HPLOT 181,49 TO 192,35
605  NEXT I
700  FOR D = 0 TO 27
710  HCOLOR= 4
712  IF D = 27 THEN 752
720  IF  INT (D / 2) <  > D / 2 THEN 750
730  HCOLOR= 3
750  HPLOT 192,32 TO 192,27 TO 193,34 TO 194,25 TO 195,
     35 TO 196,24 TO 197,36 TO 198,23 TO 199,36 TO 200,
     23 TO 201,35 TO 202,25 TO 203,34 TO 204,28 TO 204,
     32
752  HPLOT 190,24 TO 188,24 TO 189,33: HPLOT 187,21 TO
     184,28 TO 185,34: HPLOT 183,19 TO 180,27 TO 181,
     35: HPLOT 180,16 TO 175,26 TO 177,37: HPLOT 193,
     29 TO 160,22
760  NEXT D
765  IF B$ = "T" THEN 350
770  PRINT "SI SHI HUAN SI DEN HO T JIAN"
771  INPUT B$
772  IF B$ = "SHI" THEN 100
773  IF B$ = "DEN" THEN 550
774  IF B$ = "T" THEN 550
775  IF B$ = "JIAN" THEN 800
778  REN 1985,27/12
800  END
```

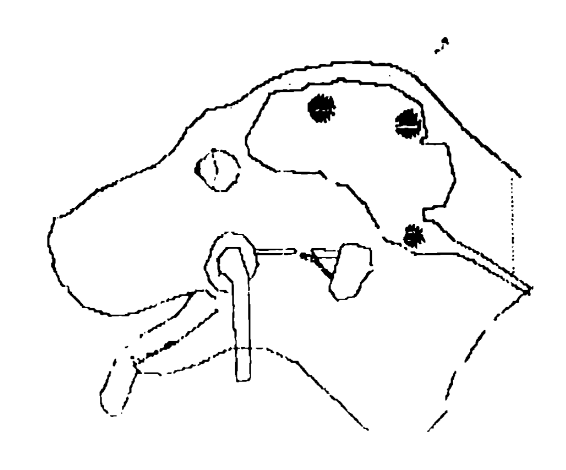


图1　结构图

上机（APPLE Ⅱ）时间简易测时程序

陈卫平

有时我们希望计算机能自动记录机器的使用时间，例如在一些教学程序中，或上机测验时。但是APPLE Ⅱ机没有关于时间的命令或函数，因此没法直接计时。那么有没有其它的办法呢？如果我们考查一下一般教学程序的使用，或上机测验的情况就会发现，计算机运行的时间事实上是很少的，绝大部分时间计算机都是在等待输入。如果把这些等待时间记录下来，即可做为使用机器的时间而不会有多大误差，根据这种情况，我们只要在计算机读入键盘输入字符时先“拦截”一下，看看有无按键按下，如果没有键按下则累计等待时间，当有键按下时，才让机器去处理读入的字符，这段记时程序存在$94B0～94EE的内存区域，其中$94B0～$94BA首先“拦截”键盘，如果没有键按下则转到$94BC～$94E3的记时程序。

$94E4～$94EE是一个子程序，用来起动该记时程序，（完成“拦截”）。

为了方便存贮和使用，程序用BASIC语言的形式存贮，第10～40行将程序存入$94B0～$94EE，第50～80行是时间初始化。其中从38048单元开始为小时，10分，分，10秒，秒，1/10秒。因此要读时间可从这些单元中读取。读数要减去176。

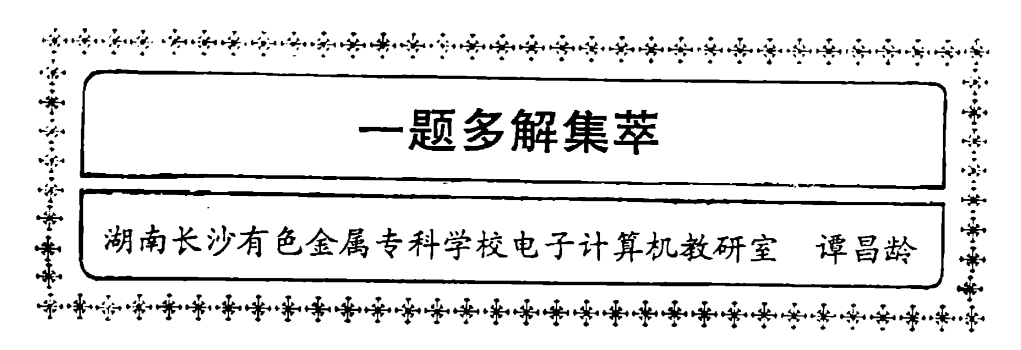
为了防止RESET之后中断记时，因此利用第90行将RESET向量指向起动子程序入口$94E4。

运行该程序后只要用语句CALL 38116或按CTRL-RESET即可起动记时程序。

最后再提醒一下，这个程序记录的时间要比实际时间少，如果你的程序运行所需时间越长，则误差越大。

```
10   FOR I = 38064 TO I + 62
20   READ J
30   POKE I,J
40   NEXT I
50   FOR I = 0 TO 7: POKE 38048 +
      I.176: NEXT
60   FOR I = 0 TO 6: POKE 38056 +
      I.186: NEXT
70   POKE 38063,219: POKE 38062.20
      0
80   POKE 38059,182: POKE 38057,18
      2
90   POKE 1010,228: POKE 1011,148:
      CALL  - 1169
95   PRINT " CALL 38116 OR CTRL -
      RESET TO START"
100  NEW
500  DATA 44,0,192,16,7,32,186,15
     9,76,27,253,96,162,7,189,168
     ,148,254,160,148,221,160,148
     ,208,8,169,176,157,160,148,2
     02,16,237,162,2,189,160,148,
     157,16,4,189,163,148,157,20,
     4,202,16,241,208,204,169,176
     ,133,56,169,148,133,57,76,19
     1,157
```

```
]
94A0- B0 B0 B0 B0 B0 B0 B0 B0
94A8- BA B6 BA B6 BA BA C8 DB
*
94B0-   2C 00 C0    BIT   $C000
94B3-   10 07       BPL   $94BC
94B5-   20 BA 9F    JSR   $9FBA
94B8-   4C 1B FD    JMP   $FD1B
94BB-   60          RTS
94BC-   A2 07       LDX   #$07
94BE-   BD A8 94    LDA   $94A8,X
94C1-   FE A0 94    INC   $94A0,X
94C4-   DD A0 94    CMP   $94A0,X
94C7-   D0 08       BNE   $94D1
94C9-   A9 B0       LDA   #$B0
94CB-   9D A0 94    STA   $94A0,X
94CE-   CA          DEX
94CF-   10 ED       BPL   $94BE
94D1-   A2 02       LDX   #$02
94D3-   BD A0 94    LDA   $94A0,X
94D6-   9D 10 04    STA   $0410,X
94D9-   BD A3 94    LDA   $94A3,X
94DC-   9D 14 04    STA   $0414,X
94DF-   CA          DEX
94E0-   10 F1       BPL   $94D3
94E2-   D0 CC       BNE   $94B0
94E4-   A9 B0       LDA   #$B0
94E6-   85 38       STA   $38
94E8-   A9 94       LDA   #$94
94EA-   85 39       STA   $39
94EC-   4C BF 9D    JMP   $9DBF
94EF-   00          BRK
```

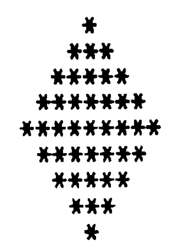


一题多解集萃

湖南长沙有色金属专科学校电子计算机教研室　谭昌龄

一题可有多解，本是一条普遍规律，而对程序设计问题则尤为如此。然而，我们在教学过程中发现，很多同学仍然只习惯于一题一解的思维模式，可他们的作业，却可以做出各种各样的千奇百怪的不同答案来。有些颇有独到之处，有些虽然正确，但方法欠佳，也有些或许是错误的。分析和评价各种不同的正确解法，对于打破学生受缚于教科书的局限性，培养发散性思维，开发创造能力，是很有益处的。下面以一题为例，集八种风格各异的不同解法，分析与评述如下：

题：用BASIC语言编写，打印下列图形的程序。



一、教科书的答案——分而治之，略嫌不足

这是教科书上的答案，是一种容易想到的办法。

程序1 ：

```
10  FOR X = 1 TO 5
20  PRINT  TAB( 6 - X);
30  FOR I = 1 TO 2 * X - 1
40  PRINT "*";
50  NEXT I
60  PRINT
70  NEXT X
80  FOR Y = 1 TO 4
90  PRINT  TAB( 1 + Y);
100  FOR I = 1 TO 9 - 2 * Y
110  PRINT "*";
120  NEXT I
130  PRINT
140  NEXT Y
150  END
```

简评：这个程序的设计思想是，苦于对循环终值和TAB函数的自变量难以找到一个统一的表达式，不得不分段处理。该程序在运行时间与占用内存方面并无浪费，且容易想到，因而具有一定的代表性。但在书写与输入时却略为麻烦一些，这是它的美中不足之处。顺便指出，有些教科书每一行图案用一个PRINT语句，一共用10个语句行（包括END语句），就可圆满地打印出题目所要求的图案。这个办法，无疑也是正确的，语句行数也并不比程序1多，但它却缺乏通用性，是一个笨办法，因而不宜提倡。

二、同学们的努力——合二而一，尚欠完美

这是学生作业中的两种解答，颇具代表性，目的是想把两段程序合而为一，这种努力是值得肯定的。

程序2 ：

```
10  FOR X = 1 TO 9
20  IF X >  = 6 THEN 50
30  PRINT  TAB( 6 - X);
40  FOR A = 1 TO 2 * X - 1: GOTO
     70
50  PRINT  TAB( X - 4);
60  FOR A = 1 TO 19 - 2 * X
70  PRINT "*";
80  NEXT A
90  PRINT
100  NEXT X
110  END
```

程序3 ：

```
10  FOR I = 1 TO 9
20  IF I >  = 6 THEN 40
30 N = 6 - I:X = 2 * I - 1: GOTO
     50
40 N = I - 4:X = 19 - 2 * I
50  PRINT  TAB( N);
60  FOR A = 1 TO X
70  PRINT "*";
80  NEXT A
90  PRINT
100  NEXT I
110  END
```

简评：前者的不足是两个循环初始语句对应着一个循环终端语句，这看起来有点别扭，虽然并不出现语法错误。后者经过改进，好了一些，但仍然没有跳出依靠条件语句转来转去的圈子。

三、老师的提示——思路各异，殊途同归

有必要由老师向学生介绍以下两种不同的常用解法，以便启发思维，提供借鉴。

程序4：

```
10  FOR I =  - 4 TO 4
20  PRINT  TAB(  ABS (I) + 1);
30  FOR J = 1 TO 9 - 2 *  ABS (I)

40  PRINT "*";
50  NEXT J
60  PRINT
70  NEXT I
80  END
```

简评：终于找到了一个统一的表达式，这一点，是应该尽力而为的。这里的关键是调整循环控制变量的变化范围。这个办法很难想到吗？不，一般说来，对称的问题总是用对称的手段来解决的。

程序5：

```
10  READ X,Y
20  IF Y =  - 1 THEN 100
30  PRINT  TAB( X);
40  FOR A = 1 TO Y
50  PRINT "*";
60  NEXT A
70  PRINT
80  GOTO 10
90  DATA 5,1,4,3,3,5,2,7,1,9,2,7,
     3,5,4,3,5,1,-1,-1
100  END
```

简评：这是一种与前不同的全新的处理方式，一般对于那些不那么规则的图案较多地采用此法。当然，也可改用循环，但采用终止标志的办法更为通用些。

四、集诸家之长——百花齐放，各有千秋

下面再提供几种采用不同的程序手段和设计思想编制的程序，以开阔视野，启迪思维。

程序6：

```
10  FOR I =  - 4 TO 4
20  PRINT  SPC(  ABS (I));
30  FOR M = 1 TO 9 - 2 *  ABS (I)

40  PRINT "*";
50  NEXT M
60  PRINT
70  NEXT I
80  END
```

简评：这个程序不用TAB函数而用SPC函数，即不是移动光标而是填充空格，二者出发点不同，结果却是一致的，真有异曲同工之妙。

程序7：

```
10 A$ = "    *********"
20  FOR I =  - 4 TO 4
30  PRINT  MID$ (A$,5 -  ABS (I),
    9 -  ABS (I))
40  NEXT I
50  END
```

简评：真是独具匠心，巧妙之极！完全冲破了原来的框框，采用字符串函数设计的奇妙独特的程序，简练而明朗。您以为如何？

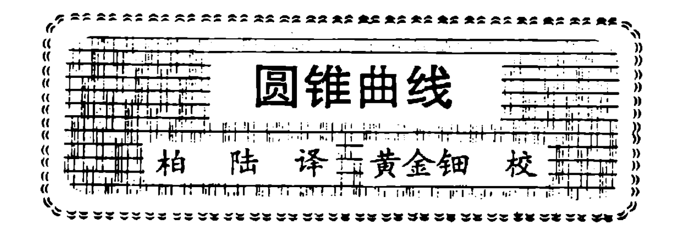
程序8：

```
10  FOR I = 1 TO 9 STEP 2
20  GOSUB 100
30  NEXT I
40  FOR I = 7 TO 1 STEP  - 2
50  GOSUB 100
60  NEXT I
70  END
100  PRINT  TAB( 6 - (I + 1) / 2)
     ;
110  FOR J = 1 TO I
120  PRINT "*";
130  NEXT J
140  PRINT
150  RETURN
```

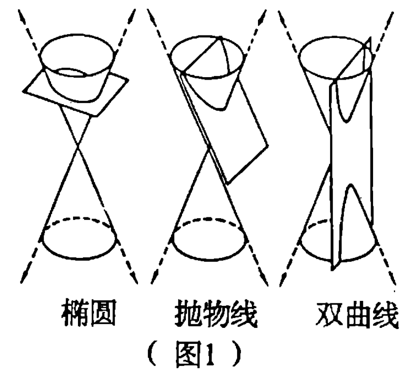
简评：这里除了用到子程序之外，您还发现了其他与前不同之处吗？前面的程序，循环控制变量变化的依据都是图形的行数，而这里依据的是每行的星号个数。这也是一种思路的革新。还可以寻找别的依据，比如每行空格的个数，这就请您自已来试一试了。

五、结　语

看了以上程序，您有何感想？是否可以起到抛砖引玉的作用，令您想入非非，产生许多更加新颖奇特的设想来呢？若能如此，即是本文作者的心愿。



圆锥曲线的命名是由于它们是平面截圆锥所得的。图1 表明一对锥面被平面截得的切面是椭圆，抛物线还是双曲线取决于此平面的倾斜程度。由图1 亦不难看出，一个点，一条直线和一对直线也可以作为圆锥的截线而得到。然而，通常把那些真正“弯曲”的截线称为圆锥曲线。



椭圆　抛物线　双曲线

（图1）

学生们可以想象在图1 中拿着平面慢慢倾斜从而改变圆锥曲线的形状。如果平面垂直于锥面的轴，其所得的圆锥曲线是一类特殊的椭圆——圆。当此平面从上述位置开始倾斜时，所得的圆锥曲线是椭圆。在图2 中，随着角θ趋近于角θ_0。椭圆的“长度”增加而“宽度”不变。

显然，当θ达到θ_0时，圆锥曲线“裂开”一个口。它不再是椭圆而是一条抛物线。当这个平面继续倾斜，它立即与一对锥面交出两条截线。此截线由两条不交的支所组成，称为双曲线。记住下面这一点是重要的：抛物线是一族椭圆和一族双曲线之间的不明确边界：

这些圆锥曲线——圆，椭圆，抛物线，双曲线——能够作为微机的很多发现性程序的课题。本文提供了两个这样的程序：“描画圆锥曲线”（表1）和“过5 点的圆锥曲线”（表2）。这两个程序使学生能够模拟把椭圆“扩充”成抛物线，然后再“扩充”成双曲线。

这些Applesoft BASIC程序包含了PRINT语句，以便向学生说明如何把自己的想法反映到程序中。

“描画圆锥曲线”

每一条圆锥曲线都有如下形式的方程

$$AX^2+BXY+CY^2+DX+EY+F=0$$

“描画圆锥曲线”（表1）的程序允许用户输入常数A，B，C，D，E和F，然后画出圆锥曲线并且请求选择另外的常数。于是，很多不同的圆锥曲线在荧屏上显示出来。

可叹的是，在很多代数教科书中，对被称为混合项的BXY没有加以讨论，以致于被讨论的圆锥曲线仅仅是其轴平行于X轴和Y轴的那一类。我想产生这种情况的原因是在B ≠0 时，“旋转角”的公式用正切函数或余切函数表示。这一点在代数课程中是经常不被讨论的，然而微机的图解计算法却能使学生们自己看到混合项的影响。

在表1 中，为了将程序缩短至最小程序和减少执行时间，我们已经删去了REM语句。可是，下列的语句可以加插进去。

```
25   REM TRANSFORMATION FORMULAS
         BETWEEN (U,V) APPLE
         COORDINATES AND
         TRADITIONAL (X,Y)
         CARTESIAN COORDINATES
335  REM TRANSFORM CONIC SECTION
         BY ROTATING
465  REM PLOT X,Y AXES AND
         CALIBRATE THEM
565  REM IF C<>0 THEN FOR EACH X,
         SOLVE FOR LOWER AND
         UPPER Y, AND PLOT THE
         CORRESPONDING
         POINTS (U,V)
715  REM IF C=0 AND A<>0 THEN FOR
         EACH Y, SOLVE FOR LEFT AND
         RIGHT X, AND PLOT THE
         CORRESPONDING
         POINTS (U,V)
865  REM IF A=0 AND C=0 AND B<>0
         THEN FOR EACH X, SOLVE FOR
         Y, AND PLOT THE ROTATED
         HYPERBOLA
965  REM IF A=0, C=0, B=0, AND
         E<>0 THEN FOR EACH X,
         SOLVE FOR Y, AND PLOT THE
         LINE
1045 REM IF A,B,C,E ALL=0 AND D<>0
         THEN PLOT THE VERTICAL
         LINE
```

下面列出的是我的学生用表1 的程序做过的实验：

1·描画$X^2+CY^2-Y=0$，其中C＝2，1，1／2，1／3，0，－1／3，－1／2，－1和－2。这对应于一个平面截一个锥面，以便从产生一个椭圆（C＞0）到突然变成抛物线（C＝0）。

2·描画$X^2-Y^2=16$，$Y^2-X^2=16$，$XY=16$和$XY=-16$。这里给出4条不同位置的双曲线，那么，这些双曲线在另外一些位置其方程又是什么？

3·描画出各种曲线对$AX^2+BY^2=36$和$AX^2-BY^2=36$，并且比较每一对曲线。然后，将它们和$BX^2+AY^2=36$，$BX^2-AY^2=36$比较。

4·描画同心圆，同心椭圆族等等。其中心不一定在（0，0）点。

5·描画一双曲线和它的渐近线（它们也是圆锥截线）。

6·插入一句

```
465 GOTO 560
```

然后，描出如图3的奥林匹克标志。

"过5点的圆锥曲线"

假定计算机用户输入平面上的五个点，希望看到过这五点的（唯一的）圆锥曲线。对于很多选出的点，该作出哪一类圆锥曲线——一个椭圆，抛物线，双曲线，或其一种退化情形，如一直线或点，这不是显而易见的。既然抛物线和其退化情形是难以确定的边界，为了简化程序，我们可以稍微随机地选择5个点，以保证过这些点的圆锥曲线是一个椭圆或双曲线。又，这种随机化将保证F′≠0时$A'X^2+B'XY+C'Y^2+D'X+E'Y+F'=0$。于是用F′除两边，产生方程形式如下$AX^2+BXY+CY^2+DX+EY+1=0$。这个仅有5个常数的方程被5个点唯一确定。这个程序可以用求逆矩阵的方法解五个5元方程组成的方程组。既然仅仅是稍微的随机化，几乎肯定其矩阵是可逆的。表2就是这样的程序。

虽然，描图输出大概是本程序的最大特色，但圆锥曲线的打印方程也给学生提供一个培养发现能力的机会。我布置给学生的一个作业有下列五个步骤：

1·执行了语句180至220的程序所提供的例子后，用特殊的数A，B，C，D和E写出下列形式的方程。

$$AX^2+BXY+CY^2+DX+EY+1=0$$

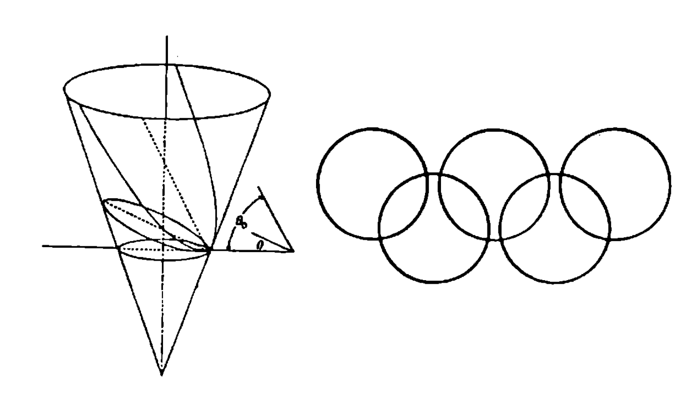
2·确定五个满足你的方程的不同的点。（对X的不同的值，如0，－1，＋1，用二次求根公式求Y，或由Y求X）

3·输入5个点，过5个点（在它们已被稍微随机化）后的圆锥曲线方程将被打印出来。如果它的系数与输入的系数几乎相等，则第二步骤是成功的。

4·若图形是椭圆，你能够仅仅移动这5个点中的一个，以得到双曲线吗？（或由双曲线得椭圆）找出输入5个点的最小变化的实验需要从一类圆锥曲线变成另一类圆锥曲线。

5·选择5个点P，Q，R，S，T，使过它们的圆锥曲线是一个椭圆，移动P到一个新的位置P，"使过P″"，Q，R，S，T的圆锥曲线是一双曲线。注意到若P，Q，R，S，T和P″都有整坐标，则圆锥曲线方程将仅有整系数。现在，想象一下P沿着一条线移至P″。与之联系的圆锥曲线开始是一个椭圆，但最后变成一双曲线。在P与P″之间的某一个特殊点P′上，此圆锥曲线是一抛物线。再用"描画圆锥曲线"的程序可找到其方程。

应该鼓励学生交替地使用这两个程序。例如，在上述作业的第3步骤后，学生们可以把方程输入"描画圆锥曲线"程序并且检验这个图与用"过5点的圆锥曲线"所给出的图是否匹配。



图二

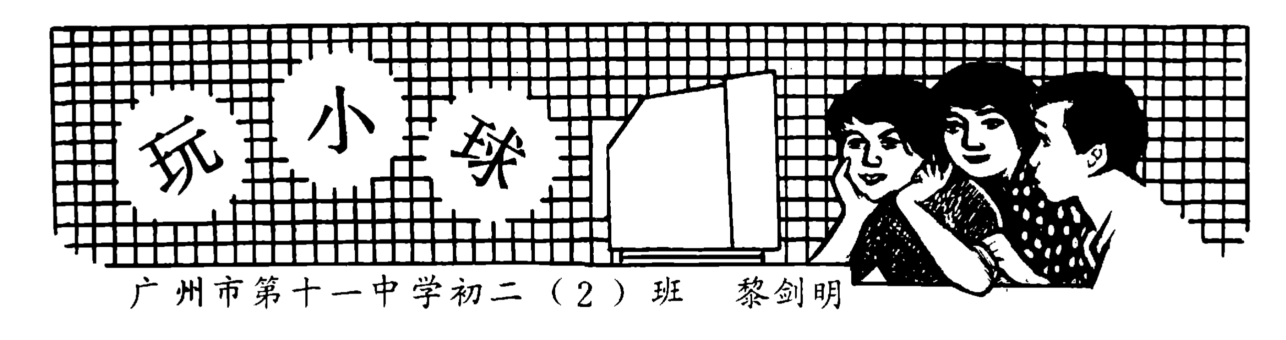
图三　奥林匹克圈，各大州的标志

表1

```
10  REM     CONIC THROUGH 5 POINTS
20  DEF  FN I(X) =  INT  (.5 + X)
30  DEF  FN Y(V) = (96 − V) / T
40  DEF  FN U(X) = 140 + S * X
50  DEF  FN V(Y) = 96 − T * Y
60 R = 4:S = 10:T = 10
70 Z9 = 1+INT (3 *  RND (1))+4 *  INT (2 *  RND
      (1))
80  HOME : PRINT "THIS PROGRAM GRAPHS
      THE UNIQUE CONIC"
90  PRINT "SECTION THAT PASSES THROUGH
      5 POINTS"
100  PRINT "WHICH YOU MAY INPUT.": PRINT
110  PRINT "IN ORDER TO SEE YOUR 5
      POINTS,"
120  PRINT "AS WELL AS THE CONIC, KEEP"
130  PRINT : PRINT "            −13 < X < 13"
140  PRINT "             −9 < Y < 9."
150  PRINT : PRINT "FOR 1ST RUN, INPUT THE
      5 POINTS IN"
160  PRINT "ONE OF THESE ";: INVERSE :
      PRINT "COLUMNS";: NORMAL
170  PRINT ":": PRINT
180  PRINT "    5,0      5,0      5,0      4,5"
190  PRINT "    0,5      0,5      0,5      5,4"
200  PRINT "   −5,0     −5,0     −5,0      2,4"
210  PRINT "    0,−5     0,−5     0,−5     4,2"
220  PRINT "    4,4      2,2      2,−2    −5,7"
230  PRINT :N = 5: FOR I = 1 TO N
240  PRINT "INPUT POINT  #"I":
      X("I"),Y("I") = ";
250  INPUT " ";X(I),Y(I)
260 X(I) = X(I) +  RND (1) / (2 * S)
270 Y(I) = Y(I) +  RND (1) / (2 * T)
280 W(I,1) = X(I) * X(I)
290 W(I,2) = X(I) * Y(I)
300 W(I,3) = Y(I) * Y(I)
310 W(I,4) = X(I):W(I,5) = Y(I): NEXT I: PRINT
320  HOME : PRINT "TO AVOID A SINGULAR
      MATRIX IN THE METH-"
330  PRINT "OD OF SOLUTION, YOUR CHOICES
      OF X,Y ARE"
340  PRINT "SLIGHTLY RANDOMIZED TO NEW
      VALUES:": PRINT
350  FOR I =  1 TO N: PRINT
      "X("I"),Y("I") = "X(I)", "Y(I): NEXT I
360  FOR I to 1 TO N
370 W(I,I) = W(I,I) + 1: NEXT I
380  FOR M = 1 TO N:P = W(M,M) − 1
390  FOR J = 1 TO N:W(M,J) = W(M,J) / P:
      NEXT J
400  FOR I = 1 TO N: IF I = M THEN 430
410 Q = W(I,M): FOR J = 1 TO N
420 W(I,J) = W(I,J) − Q * W(M,J): NEXT J
430  NEXT I: NEXT M
440  FOR I = 1 TO N:W(I,I) = W(I,I) − 1: NEXT I
450  FOR I = 1 TO N: FOR J = 1 TO N
460 Z(I) = Z(I) − W(I,J): NEXT J,I
470  PRINT : PRINT "THE CONIC THROUGH
      YOUR 5 POINTS HAS"
480  PRINT : PRINT "COEF. OF  X^2:";:  POKE
      36,17: PRINT Z(1)
490  PRINT "COEF. OF XY:";: POKE 36,17:
      PRINT Z(2)
500  PRINT "COEF. OF  Y^2:";: POKE 36,17:
      PRINT Z(3)
510  PRINT "COEF. of X:";: POKE 36,17: PRINT
      Z(4)
520  PRINT "COEF. OF Y:";: POKE 36,17: PRINT
      Z(5)
530  PRINT "CONSTANT TERM:";: POKE 36,17:
      PRINT 1
540  PRINT : PRINT "PRESS 'RETURN'.";: GET
      A$: HOME
550 A = Z(1):B = Z(2):C = Z(3):D = Z(4):E =
      Z(5):F = 1
560  HGR : POKE 49234,0: HCOLOR= 3
570  FOR I = 1 TO N
580  FOR V =  FN V(Y(I)) − 3 TO FN V(Y(I)) + 3
590  IF V < 0 OR V > 191 THEN 630
600 U1 = FN U(X(I)) − 3:U2 = FN U(X(I)) + 3
610  IF U1 < 0 OR U2 > 279 THEN 630
620  HPLOT U1,V TO U2,V
630  NEXT V: NEXT I
640  FOR V = 1 TO 190 − R STEP R
650  HCOLOR= 3
660 Y1 =  FN Y(V):Y2 =  FN Y(V + R)
670 B1 = B * Y1 + D:B2 = B * Y2 + D
680 D1 = B1 * B1 − 4 * A * (C * Y1 * Y1 + E *
      Y1 + F)
690 D2 = B2 * B2 − 4 * A * (C * Y2 * Y2 + E *
      Y2 + F)
700  IF D1 < 0 OR D2 < 0 THEN 950
710 X1 = ( − B1 −  SQR (D1)) / (2 * A)
720 X2 = ( − B1 +  SQR (D1)) / (2 * A)
730 U1 =  FN I( FN U(X1)):U2 =  FN I( FN U(X2))
740 Z1 = ( − B2 −  SQR (D2)) / (2 * A)
750 Z2 = ( − B2 +  SQR (D2)) / (2 * A)
760 W1 =  FN I( FN U(Z1)):W2 =  FN I( FN U(Z2))
770  IF U1 < 0 OR U1 > 279 THEN 800
780  IF W1 < 0 OR W1 > 279 THEN 800
790  HPLOT U1,V TO W1,V + R
800  IF U2 < 0 OR U2 > 279 THEN 830
810  IF W2 < 0 OR W2 > 279 THEN 830
820  HPLOT U2,V TO W2,V + R
830 L = 5 *  SGN (U2 − U1)
840  IF U1 < 0 THEN U1 = 0
850  IF U1 > 279 THEN U1 = 279
860  IF U2 < 0 THEN U2 = 0
870  IF U2 > 279 THEN U2 = 279
880  HCOLOR=  Z9
890  IF U1 + L < 0 OR U1 + L > 279 THEN 920
900  IF U2 − L < 0 OR U2 − L > 279 THEN 920
910  HPLOT U1 + L,V TO U2 − L,V
920  IF U1 + 2 * L < 0 OR U1 + 2 * L > 279 THEN
      950
930  IF U2 − 2 * L < 0 OR U2 − 2 * L > 279 THEN
      950
940  HPLOT U1 + 2 * L,V + R TO U2 − 2 * L,V + R
950  NEXT V: FOR J = 1 TO 2000: NEXT J: PRINT
      : PRINT
960  TEXT : HOME : VTAB 9: PRINT "TO SEE
      THIS GRAPH AGAIN, TYPE 'Y'."
970  PRINT : PRINT "TO START ANOTHER RUN,
      TYPE 'N'."
980  GET C$: IF C$ = "Y" THEN  GOTO 560
990  RUN
```

表2

```
10  REM     GRAPH MANY CONICS
20  DEF  FN I(X) = INT (.5 + X)
30  DEF  FN X(U) = (U − 140) / S
40  DEF  FN Y(V) = (96 − V) / T
50  DEF  FN U(X) = 140 + S * X
60  DEF  FN V(Y) = 96 − T * Y
70 S = 10:T = 10
80  HOME : PRINT "THIS PROGRAM GRAPHS
      EQUATIONS OF CONICS"
90  PRINT : POKE 36,2
100  PRINT "AX^2 + BXY + CY^2 + DX + EY
      + F = 0"
110  PRINT : POKE 36,15: PRINT "EXAMPLES:":
      PRINT
120  PRINT "A B C D E F   GRAPH"
130  FOR I = 0 TO 39: PRINT "_";: NEXT I:
      PRINT
140  PRINT "1   0   1   0   0 −25   CIRCLE"
150  PRINT "1   0   2   0   0 −25   ELLIPSE,
      VERT."
160  PRINT "2   0   1   0   0 −25   ELLIPSE,
      HORIZ."
170  PRINT "1   0 −1   0   0 −25
      HYPERBOLA, VERT."
180  PRINT "1   0 −1   0   0   25
      HYPERBOLA, HORIZ."
190  PRINT "0   1   0   0   0 −25
      HYPERBOLA, 45 DEG."
200  PRINT "1   0   0   0 −9   0
      PARABOLA, UP"
210  PRINT "0   0   1   9   0   0
      PARABOLA, LEFT"
220  PRINT "0   0   0 >0 >0 <0     LINE
      DX+BY+C=0"
230  PRINT "1   0 −1   0   0   0     TWO
      LINES"
240  PRINT : PRINT "YOU REALLY SHOULD TRY
      MANY OTHERS, TOO."
250  PRINT : PRINT "PLEASE INPUT
      COEFFICIENTS A,B,C,D,E,F:"
260  INPUT " ";A,B,C,D,E,F
270  HOME : HGR : HCOLOR = 3
280  IF B = 0 THEN 320
290  IF A = C THEN G = ATN (1): GOTO 310
300 G = .5 * ATN (B / (A − C))
310 U = COS (G):V = SIN (G)
320 A1 = A:B1 = B:C1 = C:D1 = D:E1 = E:F1 = F
330  IF B = 0 THEN 390
340 A = A1 * U * U + B1 * U * V + C1 * V * V
350 B = B1 * (U * U − V * V) + 2 * (C1 − A1) * U *
      V
360 C = A1 * V * V − B1 * U * V + C1 * U * U
370 D = D1 * U + E1 * V
380 E = − D1 * V + E1 * U:F = F1
390  IF C = 0 THEN 430
400 Q = D * D − E * E * A / C − 4 * A * F
410 Q1 = SQR ( ABS (Q))
420 K = − E / (2 * C)
430  IF A = 0 THEN 460
440 H = − D / (2 * A)
450 R = ABS (Q1 / (2 * A))
460 A = A1:B = B1:C = C1:D = D1:E = E1:F = F1
470  FOR U = 0 TO 279: HPLOT U,96: NEXT U
480  FOR V = 190 TO 0 STEP − 1: HPLOT 140,V:
      NEXT V
490  FOR U = 140 TO 279 STEP S: HPLOT U,96
      TO U,98
500  HPLOT 280 − U,96 TO 280 − U,98: NEXT U
510  FOR V = 96 TO 191 STEP T: HPLOT 138,V TO
      140,V
520  HPLOT 138,192 − V TO 140,192 − V: NEXT V
530  VTAB 21: PRINT "PLEASE WAIT. IF NO
      GRAPH APPEARS, IT"
540  PRINT "IS 'OFF THE SCREEN.' CURRENT
      COEFFICI-"
550  PRINT "ENTS ARE
      "A","B","C","D","E","F
560  IF C = 0 THEN 710
570  FOR U = 0 TO 279
580 X = FN X(U)
590 B1 = B * X + E
600 D1 = B1 * B1 − 4 * C * (A * X * X + D *
      X + F)
610  IF D1 < 0 THEN 690
620 Y1 = ( − B1 − SQR (D1)) / (2 * C)
630 Y2 = ( − B1 + SQR (D1)) / (2 * C)
640 V1 = FN I( FN V(Y1)):V2 = FN I( FN V(Y2))
650  IF V1 < 0 or V1 > 159 THEN 670
660  HPLOT U,V1
670  IF V2 < 0 OR V2 > 159 THEN 690
680  HPLOT U,V2
690  NEXT U
700  GOTO 1090
710  IF A = 0 THEN 860
720  FOR V = 159 TO 0 STEP − 1
730 Y = FN Y(V)
740 B1 = B * Y + D
750 D1 = B1 * B1 − 4 * A * (E * Y + F)
760  IF D1 < 0 THEN 840
770 X1 = ( − B1 − SQR (D1)) / (2 * A)
780 X2 = ( − B1 + SQR (D1)) / (2 * A)
790 U1 = FN I( FN U(X1)):U2 = FN I( FN U(X2))
800  IF U1 < 0 OR U1 > 279 THEN 820
810  HPLOT U1,V
820  IF U2 < 0 OR U2 > 279 THEN 840
830  HPLOT U2,V
840  NEXT V
850  GOTO 1090
860  IF B = 0 THEN 960
870  FOR U = 0 TO 279
880 X = FN X(U)
890  IF X = − E / B THEN 940
900 Y = ( − D * X − F) / (B * X + E)
910 V = FN I( FN V(Y))
920  IF V < 0 OR V > 191 THEN 940
930  HPLOT U,V
940  NEXT U
950  GOTO 1090
960  IF E = 0 THEN 1050
970  FOR U = 0 TO 279
980 X = FN X(U)
990 Y = − D * X / E − F / E
1000 V = FN I( FN V(Y))
1010  IF V < 0 OR V > 159 THEN 1030
1020  HPLOT U,V
1030  NEXT U
1040  GOTO 1090
1050  IF D = 0 THEN 1090
1060  FOR V = 159 TO 0 STEP − 1
1070  HPLOT  FN U( − F / D),V
1080  NEXT V
1090  HOME : VTAB 21
1100  PRINT "READY TO GRAPH YOUR NEXT
      EQUATION."
1110  PRINT "INPUT A,B,C,D,E,F:"
1120  INPUT " ";A,B,C,D,E,F
1130  GOTO 280
1140  END
```



广州市第十一中学初二（2）班　黎剑明

该程序可在LASER－310电脑上运行。当键入程序并打入“RUN”命令后，计算机会进入高分辨作图模式，并画出绿色的界线和障碍物和红色的小球。小球在屏幕上运动，如果小球撞在障碍物或界线上，则反弹出来，并发出“嘀”的响声。游戏者要按“M”键和“，”键来控制下边绿色的栏板，挡住小球，不让它出下界。但有时候，小球也会把障碍物“吃”掉，不反弹出来，使人防不胜防，所以要小心对付。如果游戏者控制不当，小球出了下界，计算机就会低鸣一声，并继续游戏，而游戏者则失去一次机会。当游戏者失去三次机会后，游戏便结束，计算机在屏幕上显示出游戏者的得分。如果游戏者的得分进入了前三名，计算机还会记下游戏者的得分和姓名。最后计算机会询问是否再玩，游戏者可按“Y”键再玩，也可按“N”键结束游戏。如果计算机配置的是单色或黑白显示器的，请将行号10中最后的COLOR2，1改为COLOR2，0。

变量说明：

J：栏板位置控制变量。最大为119，最小为1。

M：小球纵向移动控制变量。M为1时小球向下，M为－1时向上。

N：小球横向移动控制变量。N为1时小球向右，N为－1时向左。

X：小球横坐标值。

Y：小球纵坐标值。

K：积分累加变量。

W：失去机会次数。

程序说明：

行号10——55：游戏的数据初始化，并画出界线、障碍及栏板。

行号75——160：游戏的主程序。

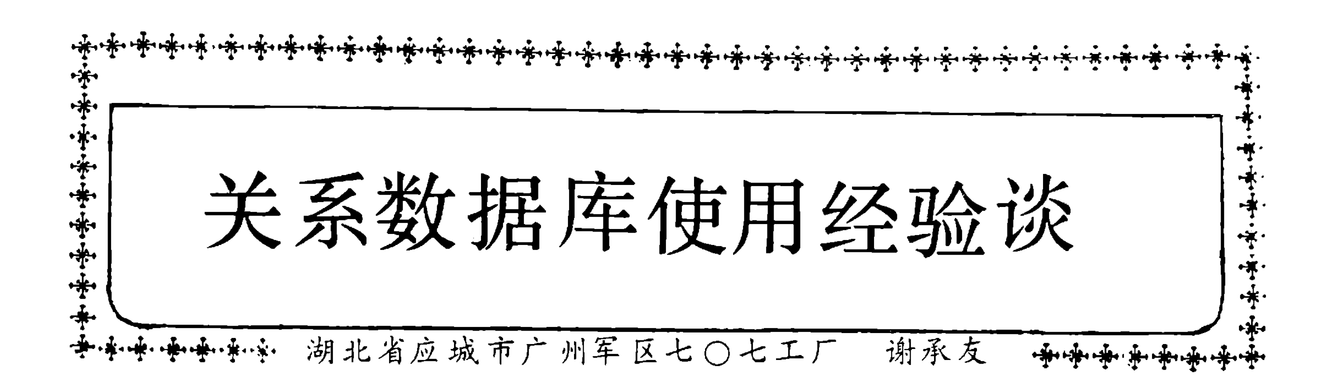
行号200——220：栏板移动子程序。

行号600——690：游戏结束后处理程序。

行号750：设置障碍物子程序。

该程序由于用了LASER－310机特有的查色命令——POINT命令，使程序大为简化，所以运行速度较快，小球和栏板移动时没有闪烁的感觉。

```
10 MODE(1):J=55:M=1:N=1:COLOR2,1
20 FOR X=0 TO 127:SET(X,0):NEXT
30 FOR Y=0 TO 63:SET(0,Y):SET(127,Y)
   :NEXT
40 FOR I=1 TO 20:SET(I,25):SET(52+I,
   25):SET(I+106,25):NEXT
45 FOR I=26 TO 35:SET(62,I):NEXT:FO
   R I=1 TO 40:GOSUB 750:NEXT
50 FOR I=0 TO 6:SET(J+I,63):NEXT
55 Y=RND(23):X=RND(126)
75 COLOR 4:SET(X,Y):COLOR 2:GOSUB 2
   00
80 IF Y=63 THEN SOUND 28,5:N=-N:W=W
   +1:IF W>2 THEN 600
100 IF N<>1 THEN IF POINT(X,Y-1)<>2
    THEN 130 ELSE 125
110 IF POINT(X,Y+1)<>2 THEN 130
125 SOUND 31,1:K=K+1:N=-N:GOSUB 750
129 GOTO 160
130 IF M<>1 THEN IF POINT(X-1,Y)<>2
    THEN 160 ELSE 150
135 IF POINT(X+1,Y)<>2 THEN 160
150 SOUND 31,1:K=K+1:M=-M:GOSUB 750
160 RESET(X,Y):Y=Y+SGN(N):X=X+SGN(M
    ):GOTO 75
200 IF INKEY$="M" THEN J=J+(J>1):SE
    T(J ,63):RESET(J+7,63):RETURN
210 IF INKEY$="," THEN J=J-(J<120):
    SET(J+6,63):RESET(J-1,63)
220 RETURN
600 CLS:PRINT"YOU GET";K;"FENS"
605 FOR I=1 TO 3
610 IF K>W(I) THEN W(I+2)=W(I+1):W(
    I+1)=W(I):W(I)=K:GOTO 640
620 NEXT I:GOTO 680
640 INPUT"PLEASE INPUT YOU NAME:";S
    $
650 S$(I+2)=S$(I+1):S$(I+1)=S$(I):S
    $(I)=S$
680 FOR I=1 TO 3:PRINT I;"---";W(I)
    ,S$(I):NEXT I
683 PRINT:PRINT TAB(7);"PLAY AGAIN
    (Y/N)"
685 K$=INKEY$:IF K$="Y" THEN W=0:K=
    0:GOTO 10 ELSE IF K$<>"N" THEN
    685
690 END
750 SET(RND(127),RND(62)):RETURN
```



关系数据库使用经验谈

湖北省应城市广州军区七〇七工厂 谢承友

目前，关系数据库——dBASE，由于它功能强、操作使用方便，因而被移植到了多种微型机上，它受到了人们的广泛应用。我在dBASE II Ver 2.4版的使用过程中，发现此版本存在着一些不容忽视的问题，今借贵刊一角，与大家共同探讨。

问题一：在dBASE状态下，对已建立的数据库进行排序或索引操作，并建立相应的新的排序文件或索引文件，以适应不同的操作需要。然而，在打开排序文件或索引文件时，监视器上显示："END OF FILE FOUND UNEXPECTEDLY."（意外地发现文件结束标志）；说明前面的排序或索引操作失败。

问题二：对已具有一定数据记录的库文件进行数据追加操作；追加结束之后，关闭库文件，然后重新打开并列表（LIST或DISP ALL）时，发现追加的新记录只能显示出几条记录或完全没有新记录。调看一下当前盘库文件（DISP FILE），所显示出的记录数增加了（增加值等于最近一次追加的记录条数）。由于不能定位于新记录，故可认为数据的追加是不成功的。

针对问题一，在DOS状态下解决较为方便，可分两步进行。

1. 将需要对其它进行排序或建立索引操作的库文件用考贝命令同盘复制成相应的排序文件或索引文件。

2. 进入dBASE状态，按以上复印的文件名对库文件进行排序或建立索引。

按上述步骤进行，问题即可解决。引起第一类问题发生的原因是：在dBASE状态下，执行SORT或INDEX命令并建立相应的新文件时，新文件目录结构中的文件起始分配单元号为OH，故文件长度亦为零，因而没有磁盘空间分配给该文件。所以导致排序或索引操作失败。

对于问题二，需要先分析一下数据不能被显示的原因，可按下述方法进行：

1. 在dBASE状态下，调看库文件结构（LIST STRU），并记下每条记录所占用的字节数（即各字段宽度之和）和记录条数，将字节数加1（因每条记录之间有一字节20H作为间隔），所得之和乘以记录条数，再将所得之积加上520（库结构记录、建立日期、文件标志等共占用数据区开始部分的520字节），即可得出库文件目前的真实数据长度。

2. 在DOS状态下启动调试程序（DEBUG.COM），用装配命令（LOAD）将磁盘的文件分配表（FAT）和目录区（逻辑扇区的01H～0CH共11个扇区）装入内存，检索（SEARCH）并显示（DUMP）库文件的目录结构，记下库文件的起始分配单元号（26～27字节），并将其转换为十进制数。然后与第一步所得的库文件目前数据真实长度相比较，可以发现，数据区实际占用长度大于目录结构中说明的文件长度。

由此可见，目录结构中的文件长度没有随数据的追加而相应加长。其原因是：库文件目前占用的扇区后面有其它文件占用的扇区或坏的扇区，当库文件记录数增加而需要增加新的分配单元时，所分配的单元号就不连续，中间间隔一个或几个扇区。在扇区号发生跳跃性变化之前，目录结构中的文件长度可随记录的增加而加长，一旦扇区号发生了跳跃性变化，目录结构中的文件长度就不变了。在磁盘读时，读入的文件长度取决于目录结构中的文件长度，因此，发生跳跃后的所有扇区数据都不能被读，所以无显示。这是dBASE IIVer 2.4

关于中文dBASE－Ⅲ INT取整函数的应用

上海市交通部第三航务工程局一公司电算站　王新华

在中文dBASE－Ⅲ中取整函数INT有其特别的地方。若想从数据库中某一数字字段作取小数点的某一位进行计数、累加，（这类运算在工资管理系统中是常用的）在尾数为零时，若单纯地用取整函数INT(X)来计算，往往会在计算算式上是对的，但结果却与希望得到的不同。

例：有某数据库名：AAA.DBF中有字段ABC，是数字型的，设指针在某一记录位置时数字是123.40。若要取小数点后二位数来进行记数或运算，如果单纯地用取整函数来计算，就会出现错误。

就上述记录字段进行上述计算，程序可写为：

```
use a: AAA
A=ABC-INT(ABC)
B=INT(A*10)
C=INT(A*100-B*10)
? A, B, C
```

希望得到的结果：A=0.4，B=4，C=0

然而在中文dBASE－Ⅲ中，运行的结果：A=0.4，B=3，C=9的结果。

若要得出正确的结果：必须在取整函数中加入一个微小量，指令应为如下：

```
A=ABC-INT(ABC)
B=INT(A*10+0.0001)
C=INT(A*100+0.0001-B*10)
```

才能得出A=0.4，B=4，C=0的排除机器零影响的正确结果。

同样也可以用子串函数来达到同样的要求：将数字段ABC中的值转换成字符串，然后再定位取数，最后再把它转换成数字，进行运算。指令可写成：

```
B=VAL(SUBSTR(ABC, 6, 2), 5, 1)
C=VAL(SUBSTR(ABC, 6, 2), 6, 1)
```

同样可以得到B=4，C=0的正确结果。

版的一个至命缺点。

为了解决这一问题，可以接着做以下几步工作：

1．根据目录结构中的起始分配单元号，查FAT中该库文件的实际分配单元数，将此数乘以1024（FAT每次分配两个扇区，每扇区长512字节），之积为库文件实际占用的磁盘空间。再将此数转换为16进制数。

2．用输入命令（ENTER）将库文件目录结构中的文件长度（28～31字节）改写为上一步所得之值。

3．在确认无误后，用写命令（WRITER）将修改后的数据存盘，返回（QUIT）到DOS状态。

值得注意的是：以上操作涉及到了磁盘的FAT和目录区，所以在做此项工作之前先做好备份，并在备份盘上操作，以免发生意外。

库文件长度进行了人为的变更，它不会影响在dBASE状态下对库文件的任何操作。这是因为在写库记录数据时，最后一条记录的最后一字节之后要写上文件结束标志“1AH”；当读库文件记录数据时，无论文件有多长，只要遇到“1AH”，读数就结束。

以上是本人在IBM-PC/XT机上应用dBASEⅡ的一点体会，难免有不妥之处，请批评指正。

微型计算机应用系统的安全技术

上海市电子振兴办公室　劳诚信

当今世界计算机已广泛应用于社会生产和生活各个领域，大大提高社会生产力和工作效率，但是如果对使用计算机的安全问题认识不足，也可能要付出昂贵的代价，影响计算机的推广使用。因此，各国政府都高度重视计算机安全与信息保护。

一、计算机应用安全的概念

计算机应用系统由硬件和软件组成，而且产生效益主要靠软件。因此，计算机系统的安全内容与一般电器设备有较大的差异：一般电器设备安全内容主要包括不发生触电、火灾、机械外伤等；而计算机系统安全除了上述内容外，主要指防止计算机系统的非授权使用，以防止计算机软件与信息的破坏和泄漏，以保护用户的利益。

具体地说，微型计算机安全技术和措施协助用户解决以下问题：

· 采取措施使电源失常和中断所造成的数据处理中断降为最低；

· 降低火灾、烟尘和水害破坏的风险；

· 提供安全运行环境，避免用户数据文件的偶然性破坏；

· 保护软盘中存放的信息文件；

· 防止未授权人员擅自使用计算机系统；

· 防止未授权人员擅自取得复制资料、修改己存入系统的数据或数据文件非法输入；

· 如果用户的计算机系统出现故障，能尽快地恢复正常运行。

计算机安全专家公认，保护计算机系统安全有四层措施：第一层是法律和社会管理，它通过国家法和地方法及社会上允许的行为方式提供保护；第二层是用户本单位的规章制度，规定有关微型计算机系统使用方法和手段，以提供保护；第三层是物理保护，使用防盗和防火技术、软盘保护等；第四层是采用硬件和软件方法提供保护。

二、法规及管理的安全措施

1·制定国家计算机安全法规

瑞典于1973年制定了数据法，是世界上第一个制定计算机安全法的国家。现在各国政府都高度重视信息的安全与保护，美国由里根总统任命一个计算机安全指导小组，总统还签署了一项计算机系统安全法。英国、法国、联邦德国、丹麦等国先后设立了部长级联合委员会，有二十多个国家制定了计算机安全法规，做到计算机安全保护有章可循，有法可依。

· 为了保护软件产品不受侵犯，国际上也建立了保护法，常用的有：专利保护法，版权保护法，商业秘密保护法等。

· 关于信息的技术规范，许多国家也正在制定，例如加拿大制订了政府用小型计算机安全标准；美国制定了政府机关计算机房标准，数据加密标准，计算机网络身份认证指南等。

2·加强计算机安全制度

· 对工作人员进行安全教育，这是改善和实施有效安全保密措施的重要前提。

· 严格规定信息资源共享的等级和范围，明确批准授权使用人员。

· 对于从事有密级信息的计算机软硬件工作人员，必须坚持先审查后录用。国外计算中心都遵循一条基本安全规定："但所被通知解雇的人员，应立即禁止接触计算机及任何有关文件"。

· 采用分工负责制，即各个不同的个人负责不同的操作步骤，因而，一个违反系统安全保密性的操作要取得成功，必须要几个人勾结才行，这就大大减少信息被窃和破坏的可能性。

· 严格规定计算机系统使用方法和操作安全，

有效地解决允许“谁”可以操作，“哪些信息”可被操作等。

三、计算机系统使用环境的安全技术

1．供电系统的安全技术

为了防止电源故障而影响计算机系统的安全性，通常可采用下述措施：

· 采用不间断供电设备，如蓄电池后备电源，当断电时接入使用，或直接由蓄电池供电。

· 采用电源调节稳压器供电，例如让50赫市电经电子交流稳压器后，再供计算机使用。电源调节器可衰减许多暂态冲击，幅度波动和电压脉冲。

· 专用供电线路供电即该供电线路不得用于任何可触产生噪音的用电设备，微型计算机系统的所有设备接入同一供电线路或电源。

· 电源要有良好的安全接地和抗干扰接地，安全接地电阻一般应小于5欧姆；抗干扰接地电阻应在2欧姆左右。

2．计算机机房环境条件

· 温度与湿度要求，机房内温度过高，则元器件性能变坏，存储器丢失数据，逻辑错误的概率会增加；温度过低，湿度太高，容易产生结露，引起器件生锈，触点接触阻抗增加。因此，机房较佳温度应是15°～25℃，相对湿度为40～70％。

· 防静电，积累在人体上的静电荷，对计算机可能造成巨大破坏。如果机房相对湿度低于40％，静电荷危险会大大增加。最有效的方法是采用防静电地毯和静电控制地板衬垫。

· 清洁防尘，微机房清洁度可在30万级左右，每升含尘度小于10000粒。某些微机装有磁盘驱动器的空气过滤器要定期清洁，按规定更换。

· 防水，机房一般不宜设在底层或地下室，以防潮防水，设备不用时应加盖防水罩。

· 防火，机房内和周围严禁堆放易燃、易爆炸物品，机房内配有足够数量的灭火设备。

· 防干扰，机房应远离强电磁场，超声波、X光等幅射源，有密级信息的计算机系统的机房应有屏蔽措施。

四、防止计算机信息破坏的安全技术

为了防止计算机信息的破坏，提高计算机系统的可靠性是一个重要问题。

· 系统设计冗余技术，当一个可靠性要求相当高的应用系统，往往需要采用冗余技术。一种常见方案是考虑双机系统：一台作为现行处理，另一台作为备份，一旦现行处理机失误，备份机就进行切换执行处理功能，而故障的机器被修复后，再投入系统作备份机。

据有关资料报导，双机系统的可靠性由下式导出，即

$$\text{双机系统稳定运行时间} = (MTBF)^2 / (2(MTTR)) + 3MTBF/2$$

其中，MTBF为单机平均无故障间隔时间；MTTR为平均修复时间。

如果每台单机MTBF为1000小时间，平均修复时间为10小时，则双机系统稳定运行时间在五年以上。

· 利用纠错码保护。检错码可在含通信线路在内的所有硬件中使用，当检出错误时进行动作复试，由硬件或软件进行以达到排除故障。如果复试不能使故障排除，则认为是固有故障，立即启动故障处理程序。

· 文件后备处理技术。为了处理固有故障造成的信息破坏，可采用将同一内容的文件定期进行转储，重份或多份保存。如果由于故障使文件受到破坏，则利用故障前的更新内容使原文件复原。在微机系统中，通常采用盘片复制后备副本。

五、防止信息泄漏的安全技术

计算机系统非授权使用包括计算机犯罪，被他人获取了有关信息，这就是信息泄漏，防止信息泄漏的安全技术有下述几种：

· 数据安全加密技术，信息密码化是防止重要信息泄漏的一种方法，它使程序或数据文件对任何未被授权但要破译这些程序或数据的变得难以理解。对信息按预定规则进行加密，借助于加密信息进行传送和存储；输出信息时，按同样的规则将信息译码后再使用。只要密码的规则不泄漏出去，信息就能有效地得到保护。

· 用于文件安全保密的数据压缩技术。数据压缩技术不但可减少程序和数据文件对存储空间的需求，而且可用于安全目的，有许多种数据压缩方法，最简单的是消零法，这项技术包括消去零或空格，或兼而有之。更复杂的数据压缩方法有格式替代法等。

· 资格检查安全技术。为了防止他人采用非法手段存取信息，可以对存取信息资格进行检查，只有检查合格者才能利用信息，以防止非授权使用。常用方法有：1．允许存取口令。所谓允许存取口令，就是在各个用户与系统之间预先规定某种“口

用字处理软件进行IBM FORTRAN等程序的全屏幕编辑

浙江医科大学计算机室　王性卫

《个人计算机与应用》1986年第5期上介绍了用dBASE的MODI COMM命令对IBM FORTRAN，COBOL程序进行全屏幕编辑。

现根据我们在实际工作中的经验和体会，介绍一种更为有效而简便的全屏幕编辑方法——用字处理软件WORDSTAR进行编辑。

开机，调入WORDSTAR后，采用非文本文件编辑功能命令N* 建立一个扩展名为·FOR或·COB的非文本文件，即可利用WORDSTAR的各种方便易操作的编辑命令进行程序的编辑与修改。

利用字处理软件进行编辑有以下一些优点：

①文件生成后即为·FOR或·COB文件不必如dBASE那样进行更名。

②编辑指令及其功能以功能菜单的形式提示出来。因此使用者只要具有一定的英语基础即可利用提示进行编辑。

③可任意选择变更TAB，这对于编辑FORTRAN等程序，尤为方便。

④可利用块操作命令（BLOCK），对程序中设定的段落进行整段的复制、移位和删除等。

⑤具有搜寻功能，在编辑特长程序时，十分有用。

⑥在对原有程序进行编辑修改后，WORDSTAR自动生成后备文件·BAK，以免编辑失误造成损失。

顺便提一下，利用WORDSTAR还可以对dBASE的命令文件·PRG，系统的批命令文件·BAT等进行编辑。

同样，在编辑中文程序时，也可用中文的WORDSTAR，其效果完全一样。

*WORDSTAR分别用D命令和N命令对文本文件（document file）和非文本文件Nondocument file）进行编辑。在编辑FORTRAN，COBOL等程序时，不能用D命令。

令”，用户利用系统时，先将自己的允许存取口令“出示”给系统，系统检查其口令是否与存储的允许存取口令一致，只有两者一致时才允许这个用户使用这个系统。因此，只要不让别人知道允许存取口令，非法用户就不能冒充正当用户使用系统，文件中的信息就不会被窃取或破坏。2·程序权限等级。计算机系统中的程序分成若干个权限等级，只有具有某个等级以上权限的程序，才许可调用某种子程序，当不具有某种特权的程序要求执行相应功能时，在检查出是不正当的要求后，该系统就给予拒绝。3·用户的权限等级。与程序一样，计算机系统的用户权限也分若干个等级，用户的权限等级是在系统登记用户时授予的，用户在利用系统的功能时，由系统校验权限等级。

六、结束语

当今人类社会，已进入了信息时代，国民经济各部门都在使用计算机进行信息存储处理。计算机信息系统已成为国家机密高度集中数据库，因此，计算机安全与国家命运有密切关系。意大利国家数据中心在六秒钟内被破坏的数据，需要六年才能恢复。我国计算机还不普及，但计算机安全问题不容忽视。我国许多重要经济部门大多用进口的计算机，有的附有外国监督使用的条款，使计算机安全无法保证。今后强调使用国产计算机，使用国内开发的应用软件，使用我国的加密方法，是计算机安全的基本条件。在加速计算机推广应用的同时，抓好计算机的安全工作，把它列为整个系统一个不可缺少的组成部份。